U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

INFORMÁTICA NA MANUTENÇÃO DE EDIFÍCIOS

Utilização de Sistemas de Identificação por RFID

LUÍS CARLOS SILVA VEIGA MARTINS

Dissertação submetida para satisfação parcial dos requisitos do grau de

MESTRE EM ENGENHARIA CIVIL — ESPECIALIZAÇÃO EM CONSTRUÇÕES

Orientador: Professor Doutor Rui Manuel Gonçalves Calejo Rodrigues

Co-Orientador: Professor Doutor Armando Jorge Miranda de Sousa

JULHO DE 2011

MESTRADO INTEGRADO EM ENGENHARIA CIVIL 2010/2011

DEPARTAMENTO DE ENGENHARIA CIVIL

Tel. +351-22-508 1901

Fax +351-22-508 1446

✉ miec@fe.up.pt

Editado por

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Rua Dr. Roberto Frias

4200-465 PORTO

Portugal

Tel. +351-22-508 1400

Fax +351-22-508 1440

✉ feup@fe.up.pt

🌐 http://www.fe.up.pt

Reproduções parciais deste documento serão autorizadas na condição que seja mencionado o Autor e feita referência a *Mestrado Integrado em Engenharia Civil - 2010/2011 - Departamento de Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, 2010*.

As opiniões e informações incluídas neste documento representam unicamente o ponto de vista do respectivo Autor, não podendo o Editor aceitar qualquer responsabilidade legal ou outra em relação a erros ou omissões que possam existir.

Este documento foi produzido a partir de versão electrónica fornecida pelo respectivo Autor.

À minha Mãe e ao meu Pai.

"A grande sabedoria, penso eu, é ter um sentido relativizado de tudo. Não dramatizar nada."
José Saramago

AGRADECIMENTOS

Agradeço às seguintes pessoas pelo apoio à realização deste trabalho:

Ao meu Orientador Professor Rui Calejo pelo incentivo, ideias e apoio em toda a fase deste trabalho.

Ao meu Co-Orientador Professor Armando Sousa que, apesar de não estar ligado à engenharia civil, demonstrou grande interesse no projecto e me apoiou sempre que necessitei.

Ao Doutor Simão Rocha que, ao serviço da empresa Telemax - Telecomunicações e Electrónica, Lda, se disponibilizou a partilhar o seu conhecimento no tema assim como cedeu o equipamento utilizado para a realização deste trabalho.

Ao João Araújo pelo apoio e companheirismo demonstrado no decorrer deste trabalho.

Aos meus pais por tudo.

RESUMO

Com a evolução da complexidade e tamanho dos edifícios a tarefa de catalogar informações de projecto, construção e utilização de todos os elementos e componentes presentes num edifício torna-se difícil. Muitas informações são perdidas pelas mais variadas razões, quer por não serem relatadas por escrito quer pelo seu excesso, facto que dificulta o seu acesso, sendo portanto consideradas "perdidas" pois não estão à disponibilidade de quem necessita delas numa dada altura. Este facto motiva uma maior demora e custo de manutenção na fase de utilização do edifício.

Esta dissertação tem como objectivo criar um sistema que permita o acesso fácil e directo a informações importantes à manutenção dos elementos e componentes de um edifício, permitindo a alteração e registo de ocorrências, dando conhecimento do histórico dos elementos e componentes, facilitando assim a sua manutenção e portanto melhorando o seu desempenho.

Hoje em dia, quando se fala em tratar grandes quantidades de informação, permitindo um fácil acesso, obrigatoriamente se terá que falar em informática. Neste trabalho elabora-se um *software* que permite comunicar com os elementos e componentes construtivos através da tecnologia RFID, acedendo à informação que está contida numa base de dados remota cuja estruturação constitui também um desenvolvimento deste trabalho.

Tiram-se conclusões da adaptação deste sistema à manutenção de um edifício, identificando pontos positivos, negativos e aspectos a desenvolver futuramente.

PALAVRAS-CHAVE: Manutenção, RFID, Base de dados, Software, Elementos e Componentes da Construção.

ABSTRACT

With the evolution of building's complexity and size, the task of cataloging design, project and construction information becomes harder. Many of that information is lost either because it is not written or because information overload that hampers their access, making it "lost information" because it is not available to those who need it at some point. Because of this, a higher cost and a further delay in maintenance are identified in the use phase of the building.

This dissertation aims to create a system that allows the easy and direct access to maintenance information of the building's elements and components, allowing maintenance record, giving the operator the element's historical and therefore improving its performance.

Nowadays when it comes to deal with large amounts of information, allowing easy access to it, information technology must be used. In this dissertation will be created an application and a data base that allows to communicate with the building, with the help of RFID technology.

Conclusions are drawn from the adaption of this system to the building maintenance, identifying strengths, and negative aspects to develop further.

KEYWORDS: Maintenance, RFID, database, Software, building elements.

ÍNDICE GERAL

AGRADECIMENTOS I
RESUMO III
ABSTRACT V

1 INTRODUÇÃO 1
1.1. MOTIVAÇÃO 1
1.2. ÂMBITOS E OBJECTIVOS 1
1.3. ESTRUTURA DA DISSERTAÇÃO 2
2 CONTEXTUALIZAÇÃO 3
2.1. CONSCIENCIALIZAÇÃO 3
2.1.1. CONSCIENCIALIZAÇÃO HISTÓRICA 3
2.1.2. CONSCIENCIALIZAÇÃO ECONÓMICA E SOCIAL 5
3 MANUTENÇÃO DE EDIFÍCIOS 7
3.1. GESTÃO DE EDIFÍCIOS 7
3.2. MANUTENÇÃO DE EDIFÍCIOS 8
3.2.1. CONSIDERAÇÕES INICIAIS 8
3.2.2. DEFINIÇÃO 8
3.2.3. TIPOS E ESTRATÉGIAS DE MANUTENÇÃO 9
3.3. DADOS DE MANUTENÇÃO A IMPLEMENTAR NA BASE DE DADOS 11
4 RFID 13
4.1. CONTEXTUALIZAÇÃO HISTÓRICA 13
4.2. CONSIDERAÇÕES INICIAIS 16
4.3. TAG 17
4.3.1. *TAGS* PASSIVOS 18
4.3.2. *TAGS* ACTIVOS 18
4.3.3. *TAGS* SEMI-PASSIVOS 18
4.3.4. TAGS SEMI-ACTIVOS 18
4.4. LEITORES/ANTENAS 18
4.5. SOFTWARE 19
4.6. FREQUÊNCIAS DE OPERAÇÃO RFID 19

4.7. REGULAMENTOS E NORMAS 20
5 PROPRIEDADES DO SISTEMA RFID 21
5.1. CENÁRIO IDEAL 21
5.2. ASSOCIAÇÃO DO CENÁRIO À REALIDADE TECNOLÓGICA 22
5.3. ANÁLISE ECONÓMICA 22
5.4. CONDIÇÕES LIMITE DE LOCALIZAÇÃO DOS TAGS 22
5.4.1. CONDIÇÃO MENOR 23
5.4.2. CONDIÇÃO MAIOR 23
5.4.3. CONDIÇÃO INTERMÉDIA 23
5.4.4. CONCLUSÃO 24
5.5. MEMÓRIA INTERNA E/OU BASE DE DADOS 24
5.6. LEITORES E TAGS 24
5.7. IMPLEMENTAÇÃO 27
5.8. CONCLUSÕES 27
6 SOFTWARE 29
6.1. CONSIDERAÇÕES INICIAIS 29
6.2. VISÃO DE SOFTWARE IDEAL 29
6.2.1. VISÃO DO OPERADOR 30
6.2.2. VISÃO DO GESTOR DE MANUTENÇÃO 31
6.2.3. VISÃO DO UTILIZADOR DO EDIFÍCIO 32
6.3. CONCLUSÕES 32
7 CONCEPÇÃO DO SISTEMA 33
7.1. INTRODUÇÃO 33
7.2. SISTEMA RFID 34
7.3. BASE DE DADOS 34
7.3.1. INTRODUÇÃO 34
7.3.2. ENTIDADES 35
7.3.3. ASSOCIAÇÕES 35
7.3.4. CARDINALIDADE 36
7.3.5. ATRIBUTOS 37
7.3.6. MODELO ENTIDADE-ASSOCIAÇÃO 40
7.3.7. SISTEMA DE GESTÃO DE BASE DE DADOS 41
7.4. SOFTWARE 45

7.4.1. INTRODUÇÃO AO LAZARUS 45
7.5. MAINTECH – APRESENTAÇÃO 47
7.5.1. MODO DO PROGRAMA 48
7.5.2. LEITURA DE ELEMENTO ATRAVÉS DE RFID 49
7.5.3. REGISTO 50
7.5.4. EDIÇÃO DE REGISTOS 52
7.5.5. MANUTENÇÃO 53
7.5.6. ACÇÃO 54
7.5.7. NOVO REGISTO 56
7.5.8. OUTRAS PESQUISAS 56
7.5.9. MENU ADICIONAR 58
7.5.10. MENU AVANÇADO 65
7.6. CONCLUSÕES 71
8 CONCLUSÕES 73
8.1. CONSIDERAÇÕES FINAIS 73
8.2. DESENVOLVIMENTOS FUTUROS 74

REFERÊNCIAS BIBLIOGRÁFICAS 77

ANEXOS CONSIDERAÇÕES 79
ANEXO 1 FICHA TÉCNICA MOTOROLA MC3090-Z 81
ANEXO 2 FICHA TÉCNICA SYSTEM LIMITED CS 101 85
ANEXO 3 FICHA TÉCNICA AVERY DENNINSON AD-223 87

ÍNDICE DE FIGURAS

Fig. 1 - Estrutura fundamental da gestão de edifícios. 8
Fig. 2 - Exemplo de um código de barras. 13
Fig. 3 - Exemplo de um QR Code com a informação em texto do título deste trabalho. 14
Fig. 4 - Esquema do funcionamento de um sistema RFID. 17
Fig. 5 - Vários formatos de tags, respectivamente: Moeda, Cartão plástico, Autocolante, Tag para ferro e Tag em forma de prego. 17
Fig. 6 - Exemplo de uma Polarização Circular e Linear. 19
Fig. 7 - Leitor e PDA Motorola MC3090-Z. 25
Fig. 8 - Leitor e PDA da System Limited ref. CS 101. 26
Fig. 9 - Formato da antena do *tag* da Avery Denninson ref. AD-223. 27
Fig. 10 - Esquema de funcionamento do sistema. 33
Fig. 11 - NEXUS ref.BMU-01 Leitor usado neste projecto. 34
Fig. 12 - Entidades e Associações. 36
Fig. 13 - Cardinalidade entre Associações. 37
Fig. 14 - Modelo Entidade-Associação completo. 41
Fig. 15 - Ecrã de administrador do software PostgreSQL. 42
Fig. 16 - Ecrã de Lazarus em modo de programação. 45
Fig. 17 - Logotipo do software RFID MainTech. 47
Fig. 18 - Ecrã principal do programa MainTech. 48
Fig. 19 - Menu superior em modo Operador do programa MainTech. 48
Fig. 20 - Menu superior em modo Gestor do programa MainTech. 48
Fig. 21 - Esquema de funcionamento da Leitura por RFID. 49
Fig. 22 - Ecrã principal após elemento lido (pasta Empresa Fabricante) do programa MainTech. 50
Fig. 23 - Pasta de Registo do programa MainTech. 52
Fig. 24 - Pasta Editar Registos do programa MainTech. 52
Fig. 25 - Pasta Manutenção com acções periódicas do programa MainTech. 53
Fig. 26 - Pasta Manutenção acções não periódicas do programa MainTech. 54
Fig. 27 - Pasta Acção para um exemplo de correcção do programa MainTech. 55
Fig. 28 - Menu Novo do programa MainTech. 56
Fig. 29 - Pasta Novo Registo do programa MainTech. 56
Fig. 30 - Pasta Outras Pesquisas modo Operador do programa MainTech. 57
Fig. 31 - Pasta Outras Pesquisas modo Gestor do programa MainTech. 57
Fig. 32 - Menu superior Adicionar e submenus do programa MainTech.. 58
Fig. 33 - Pasta Adicionar Empresa do programa MainTech.. 58
Fig. 34 - Organograma da criação de empresas. 59
Fig. 35 - Pasta Adicionar Categoria do programa MainTech.. 59
Fig. 36 - Pasta Adicionar LIPS do programa MainTech.. 60
Fig. 37 - Organograma de obtenção de novo *cod_lips*. 61
Fig. 38 - MainTech pasta Editar Categoria. 62
Fig. 39 - Organograma de preenchimento de tabelas na pasta Editar LIPS 63
Fig. 40 - Querys da adição e remoção de processos de manutenção a categorias. 63
Fig. 41 - MainTech pasta Adicionar Elemento. 64
Fig. 42 - Organograma e *query* da pasta Adicionar Elemento. 64
Fig. 43 - Pasta Editar Empresa do programa MainTech. 65
Fig. 44 - Menu Avançado e submenus do programa MainTech. 65
Fig. 45 - Pasta Verificação de Elementos do programa MainTech. 66
Fig. 46 - Organograma de funcionamento do ciclo de Verificaçoes dos Elementos 67
Fig. 47 - Pasta Consulta à Base de Dados do programa MainTech, exemplo de consulta de Manutenções. 68
Fig. 48 - Pasta Registo de Manutenções do programa MainTech (selecção da data). 69
Fig. 49 - Pasta Registo de Manutenções resultados do programa MainTech. 69
Fig. 50 - Organograma da verificação de garantias. 70
Fig. 51 - Pasta Garantias do programa MainTech, exemplo de output. 71

ÍNDICE DE TABELAS

Tabela 1 - Tabela de Entidades, Atributos, Tipos de dados e Constrições. 44

SÍMBOLOS E ABREVIATURAS

EFM – Elementos Fonte de Manutenção

RFID – Radio-Frequency Identification

SPAB – Society for the Protection of Ancient Buildings

IFMA – International Facility Managment Association

Euro FM – European Facility Managment

APFM – Associação Portuguesa de Facility Managment

INE – Instituto Nacional de Estatística

IFF – Identify Friend or Foe

EUA – Estados Unidos da América

LF – Low Frequency

HF – High Frequency

UHF – Ultra High Frequency

IBM – International Business Machines

EPC – Electronic Product Code

ROM – Read Only Memory

RAM – Read Access Memory

WORM – Write Once Read Many

AVAC – Aquecimento, Ventilação e Ar Condicionado

USB – Universal Serial Bus

PDA – Personal Digital Assistant

LIPS – Limpeza, Inspecção, Pró-Acção e Substituição

NIF – Número de Identificação Fiscal

SGBD – Sistema Gestor de Base de Dados

BSD – Berkeley Software Distribution

PK – Primary Key

FK – Foreign Key

IDE – Integrated Development Environment

RAD – Rapid Application Development

1
INTRODUÇÃO

1.1. MOTIVAÇÃO

A importância dada à manutenção de edifícios tem evoluído nos últimos anos. Em virtude disso, o custo de um edifício deixa de ser apenas o custo de construção mas passa também a considerar o custo necessário para manter o edifício com a qualidade desejada durante os anos que foi projectado. Esta evolução deve-se, em parte, à adopção de noções e procedimentos habitualmente utilizados no sector industrial Tome-se como exemplo o facto de hoje em dia as oficinas de automóveis, para a manutenção de um veículo, praticamente comunicarem com ele. Com a ajuda de novas tecnologias consegue-se, em tempo real, saber as mais variadas condições e históricos de componentes do automóvel não desperdiçando recursos ao procurar conhecer o problema manualmente. Quando se fala num automóvel pode-se também falar dum equipamento industrial onde, em tempo real, se consegue saber qual a produção, performance e estado das máquinas.

Porque não fazer o mesmo com um edifício?

Torna-se evidente que, também para os edifícios privados e públicos, exista uma necessidade clara de comunicar em tempo real com o eles para dessa forma obter um conhecimento mais célere do estado dos componentes do edifício, facilitando a respectiva manutenção e consequentemente contribuindo para uma redução dos custos que lhes estão associados.

1.2. ÂMBITOS E OBJECTIVOS

Aliando o conhecimento de manutenção adquirido no Mestrado Integrado de Engenharia Civil e o gosto por informática do autor, desenvolver-se-á um sistema que permita essa comunicação entre o edifício e o usuário.

Para catalogar, de uma forma não computorizada, todos os Elementos Fonte de Manutenção (EFM) de um edifício gastar-se-iam muitos recursos (tempo, papel, arquivos, etc.). Estando esta informação disponível apenas em papel, torna-se difícil o seu acesso no dia-a-dia, pois tal facto dificulta a sua pesquisa e consulta, o que limita algumas vantagens que essas informações possam oferecer. Por outro lado, muitos edifícios, com equipas de manutenção profissionais, já têm bastante informação computorizada, mas geralmente dispersa em vários ficheiros, divididos por tipos de componentes, de informação não normalizada e sem correlação entre eles.

Sendo este um trabalho pioneiro, é de salientar que se procurará fazer um "*starter job*", no qual as funções de comunicação com o edifício sejam as mais básicas, tentando então fazer uma prova de

conceito sobre uma tecnologia que, mais tarde após optimização, poderá ser implementada em edifícios para uso real.

Para tal, ir-se-á estudar o uso de RFID em componentes passíveis de manutenção, conferindo-lhes a propriedade de comunicação com a pessoa responsável pela manutenção.

O principal objectivo é provar que é possível de facto comunicar com o edifício, em que um elemento terá uma identificação única numa base de dados, e que, quando a pessoa responsável pela manutenção quiser informações sobre um elemento, vai ter acesso a elas em tempo real. Mais tarde poderá normalizar-se a informação, dando assim a possibilidade de um software ajudar nos planos de manutenção do edifício. Estuda-se também a hipótese de implementar esta tecnologia não só na construção do edifício como também na sua fase de utilização, e até nas indústrias de componentes da construção.

Pode-se sintetizar este trabalho nas seguintes "questões de investigação":

- É possível encontrar uma forma de comunicação entre um edifício e um sistema de manutenção?
- Qual a informação que um Elemento Fonte de Manutenção deve ter associado?
- Quais as vantagens que esta forma de comunicação pode trazer ao plano actual da manutenção de edifícios?
- Actualmente será um sistema viável técnica e economicamente?
- Quantos e quais os elementos de um edifício que deveriam ter esta propriedade?
- Quais serão os pontos-chave para uma adopção generalizada deste sistema ser possível?

1.3. ESTRUTURA DA DISSERTAÇÃO

O trabalho está dividido em 8 capítulos. No capítulo 1 (o presente) abordam-se as motivações, âmbitos e objectivos que o autor considerou para a execução deste trabalho. No capítulo 2 trata-se da contextualização em que se está inserido o trabalho. Nos capítulos 3 e 4 tratam-se das noções necessárias para a execução do trabalho. Neles aprofundam-se noções de manutenção, assim como se explica o funcionamento e as características de um sistema RFID.

Nos capítulos 5 e 6 o autor expõe a sua opinião sobre o que seria um sistema perfeito de comunicação entre o edifício e a equipa de manutenção assim como um software a ele associado.

No capítulo 7 demonstra-se uma solução real e testada pelo autor. Nesta solução procede-se à apresentação do sistema RFID utilizado assim como a execução de um software e de uma base de dados que trabalhem em conjunto com o sistema RFID.

No capítulo 8 tiram-se conclusões sobre o sistema utilizado, as suas vantagens, assim como os desenvolvimentos futuros previstos pelo autor.

2
CONTEXTUALIZAÇÃO

2.1. CONSCIENCIALIZAÇÃO

2.1.1. CONSCIENCIALIZAÇÃO HISTÓRICA

Desde que o Homem deixou o seu estilo de vida nómada, associado ao primórdio da agricultura para a criação de gado, começou aos poucos a fixar-se num local, como também a preocupar-se com o estado de conservação dos seus abrigos. Pode dizer-se que a manutenção de edifícios tem origem desde essa altura pois, visto que os abrigos eram fixos, obrigatoriamente utilizavam técnicas para os manter nas condições necessárias.

Efectivamente, a primeira referência encontra-se na descoberta de uma classe social no Antigo Egipto, *artífices*, que tinham como trabalho a reparação de edifícios e templos. Outra referência de manutenção de edifícios é encontrada na publicação de *Marcus Vitruvius Pollio*, na época do Imperio Romano. Referências à manutenção de edifícios surgem ainda abundantemente durante o Renascimento, com o desenvolvimento das construções militares e apalaçadas. Os Renascentistas demonstravam grande preocupação com os seus edifícios, tendo sempre em conta os cuidados com a conservação dos mesmos e os custos associados. [1]

No século XVII, na sequência de um grande incêndio em Londres, foi publicado um documento a pedido do Rei, *Building Act of London*, para impor regras na reconstrução dos três quartos da cidade consumida pelas chamas. Neste documento referenciam-se pela primeira vez medidas de manutenção como a limpeza dos vazadouros ou a manutenção do reboco dos edifícios. [1]

No século XVIII, com a revolução industrial e a utilização do ferro nas estruturas, recorreu-se à aplicação de produtos anticorrosivos para manutenção dessas estruturas. Os produtos anticorrosivos são provavelmente os primeiros produtos no mercado da manutenção de edifícios. [1]

A 20 de Agosto de 1721, D. João V através de um alvará régio[1], cria a primeira legislação em Portugal referente à protecção de monumentos históricos, a qual foi atribuída à Real Academia Histórica. [2]

O uso do termo “manutenção de edifícios” foi, contudo, apenas formalmente utilizado no século XIX.

[1] O Alvará Régio é um termo jurídico antigo usado para designar um edito real. Pode ser interpretado e caracterizado como uma licença real ou decreto régio num estado tipicamente absolutista, de uma monarquia ou de um império. Durante o período do Brasil colonial e em vários períodos da história de Portugal, os reis ou regentes fizeram uso deste edito para governar.

Em 1877 funda-se a *Society for the Protection of Ancient Buildings* (SPAB) onde a manutenção adquiriu finalmente um estatuto próprio, desenvolvendo-se progressivamente como uma área de conhecimento independente e essencial [3].

A SPAB opunha-se a violentas remodelações a que eram sujeitos edifícios medievais por parte de arquitectos vitorianos, sendo ainda hoje um reconhecido grupo de pressão britânico que luta pela preservação de edifícios antigos. Em 2006, a SPAB deu início a um novo e ambicioso projecto, denominado *Faith in Maintenance*. A sensibilização de toda a sociedade civil é uma das principais causas da SPAB, que promove ainda a *National Maintenance Week*. Entre outras iniciativas de relevo, esta associação ministra cursos de formação gratuitos a voluntários, no domínio da manutenção de edifícios. A emergência do conceito de manutenção, tal como hoje é conhecido, emanou de um movimento amplo que pugnava pela conservação do património histórico, mais até do que pela preservação de construções recentes. [4]

A Carta de Atenas é o manifesto urbanístico resultante do IV Congresso Internacional de Arquitectura Moderna, realizado em Atenas em 1933. [5]

No Congresso de Atenas foram aprovadas sete resoluções principais, denominadas *Carta del Restauro*, entre elas a necessidade de manutenção regular dos monumentos, salvaguardar obras históricas e artísticas do passado, possibilidade de reutilização dos monumentos e a importância de uma análise e documentação prévia que fundamente as intervenções e que sirva de apoio a um correcto diagnóstico. [6]

Nos anos 40, com o desenvolvimento do sector comercial da aviação, nasce a Engenharia da Manutenção, visando garantir o melhor desempenho e segurança dos aviões, através da implementação de medidas preventivas e inspecções periódicas de verificação. [7]

Nos Estados Unidos a génese da manutenção teve como palco o período pós 2ª Guerra Mundial, com a implementação de rotinas de inspecção de modo a optimizar a indústria de armamento relativamente aos seus equipamentos e, mais tarde, aos edifícios industriais. [1]

No século XX dão-se enormes progressos no domínio da manutenção e conservação de edifícios. A arqueologia é a ciência a quem cabe numa fase inicial liderar este processo, assistindo-se progressivamente a um aproximar à concepção multidisciplinar actual, integrando a arquitectura e engenharia. Em 1970 cria-se o *International Institute for Conservation – Canadian Group* em que o código de ética dava instruções aos "conservadores" para que na manutenção de edifícios culturais, se usassem materiais e técnicas que não pusessem em risco a integridade física e cultural do edifício. [1]

As metodologias de manutenção só adquirem sistematização após as consequências duma fase inicial de indústria sem manutenção. É nas instalações militares da II Grande Guerra que, nos Estados Unidos da América, se implementam rotinas de manutenção industrial. Estas rotinas foram também implementadas nos próprios edifícios, o que se revelou eficaz. Em 1961 publicou-se o *Factories Act*, onde se abordam os edifícios industriais e as condições de utilização dos operários. Em 1964, no mesmo país, publicou-se a primeira norma sobre manutenção, a BS3811, que trata de manutenção industrial, onde se incluem muitos dos conceitos usados hoje em edifícios. [1]

Vários estudos foram feitos entretanto, incidentes no custo de utilização dos edifícios, mas foi em 1980 que o Facility Management nasceu nos Estados Unidos da América. Em 1982, dada a importância do tema, criou-se uma associação internacional, *International Facility Management Association* (IFMA). Na Europa, em meados de 1990, surge a Euro FM (*European Facility Management Network*), assumindo-se como uma instituição composta por mais de 60 organizações, todas elas especializadas e centradas no campo do *Facility Management*. Estas organizações têm a sua

sede em 15 países europeus e representam fundamentalmente associações nacionais, centros educativos e de pesquisa e organizações comerciais. O objectivo desta instituição é facilitar o intercâmbio de informações, conhecimentos e experiencias entre académicos e pesquisadores do assunto. A nível nacional, o *facility management* é um domínio muito recente. Foi apenas no final do ano de 2006 que se constituiu a Associação Portuguesa de Facility Management (APFM), iniciando a sua actividade em 2007 e sendo consolidado o conceito e a organização das actividades em 2008. [7]

A APFM é uma Associação Nacional sem fins lucrativos cujos objectivos são o desenvolvimento, a investigação e a divulgação da área profissional denominada *Facility Management*. Esta visa uma gestão integrada dos locais e ambientes de trabalho, com o objectivo de optimizar os espaços, os processos e as tecnologias envolventes, sendo esta, vocacionada para todos aqueles que se dedicam à gestão de imóveis, aos que têm esta função subcontratada e para os que prestam serviços relacionados com esta área. Surgiu como iniciativa de alguns profissionais e de algumas empresas de referência no sector em Portugal que criaram esta Associação, independente, de modo a desenvolver a profissão de *Facility Manager* assim como promover o contacto entre profissionais do sector. Esta profissão é reconhecida em cerca de cinquenta países com mais de vinte mil profissionais. [18]

Em face do exposto é possível antever uma evolução ainda grande da Manutenção de Edifícios para a qual se entende poder contribuir com este trabalho.

2.1.2. Consciencialização Económica e Social

Como referido anteriormente, desde o fim do período nómada que o ser humano tenta ter cuidados de conservação dos seus abrigos. Mas até que ponto se reflecte na sociedade de hoje a necessidade de conservar os seus edifícios? Efectivamente, hoje em dia é comum ver edifícios visivelmente degradados, quer se trate de edifícios de habitação quer de edifícios públicos.

O cidadão comum desconhece as grandes vantagens que uma boa manutenção de edifícios lhe pode trazer.

A habitação é um bem que toda a população deseja obter, estando disposta a despender capital, endividando-se com empréstimos para satisfazer o custo inicial da aquisição da habitação. O problema reside em que não se dá a devida atenção aos gastos com o edifício na sua fase de utilização, muitas vezes deixando o custo da manutenção para segundo plano.

Efectivamente, uma boa manutenção faz com que se optimize a relação custo/performance do edifício na sua conservação. Esperar em demasia para reparar uma patologia fará com que os custos dessa reparação se elevem, muitas vezes para patamares de custos bem superiores aos que se apurariam se fosse executada uma intervenção na devida altura.

No caso de alguns edifícios públicos já não se nota tanto este efeito, não só porque o dinheiro não é tão difícil de obter, mas também por existirem equipas especializadas a cargo da manutenção do edifício. Infelizmente Portugal atravessa uma grave crise económica razão pela qual estes edifícios possam vir a sofrer grandes cortes económicos.

A informatização da manutenção de edifícios pode melhorar vários aspectos, quer a nível social como industrial. Com a crescente informatização da manutenção de edifícios, passaremos a ter conhecimento do comportamento em utilização dos mais variados elementos num edifício. Se uma fábrica tiver acesso a essas informações, poderá descobrir falhas nos seus produtos, e assim melhorá-los progressivamente. Com o acesso a informação de vários edifícios com soluções idênticas e/ou produtos da mesma fábrica, é possível fazer estimativas e previsões de manutenção, prevendo

estatisticamente o momento em que deve ser efectuada a manutenção, o tipo e mesmo os custos associados a esta.

As fábricas podem aproveitar toda a informação gerada pelos vários edifícios que usam os seus produtos, não só para o melhoramento dos mesmos, mas para elevarem a sua reputação e confiança do mercado com as informações de boa performance dos produtos.

Uma fábrica que inclua suporte informático para a manutenção dos seus produtos torna as suas garantias mais aliciantes e confiáveis, ou seja, o cliente sente-se mais seguro quando escolhe os produtos dessa fábrica.

O possível conhecimento do comportamento em utilização de um produto e dos respectivos custos de manutenção faz com que o comprador tome decisões mais acertadas. E da mesma forma, uma fábrica que utilize este sistema torna-se mais transparente aliciando assim mais compradores.

A informatização da manutenção de edifícios poderá fazer com que, num futuro próximo, possa existir uma base de dados nacional/internacional para comportamentos de diversos materiais de construção, permitindo que o comprador faça escolhas mais acertadas e que as fábricas se preocupem cada vez mais com a qualidade dos seus produtos, acabando por beneficiar ambas as partes. Torna-se também mais palpável a justificação da diferença de preços de um produto de qualidade superior a um produto que não tenha essa qualidade.

3
MANUTENÇÃO DE EDIFÍCIOS

3.1. GESTÃO DE EDIFÍCIOS

Antes de falar da gestão de edifícios convém esclarecer o que é exactamente Gestão. Entende-se por gestão como sendo o "conjunto de medidas de administração (de uma organização, empresa, etc.) aplicadas durante um determinado período; modo de gerir" ou "utilizar racionalmente recursos em função de um determinado projecto ou de determinados objectivos".

A gestão é amplamente usada no meio industrial, onde tem como principal objectivo optimizar processos para aumentar lucros. Do mesmo modo que é utilizada para processos industriais, é também utilizada nas mais variadas áreas tais como:

- Gestão de stock;
- Gestão de recursos humanos;
- Gestão financeira;
- Gestão de projecto;
- Gestão de património.

É portanto compreensível que para qualquer actividade que englobe um conjunto de acções e procedimentos seja possível uma gestão no âmbito da qual se optimizem esses processos. Cada área da gestão tem que ter o seu objectivo definido, tal como o lucro para a gestão industrial.

Posto isto, é portanto natural que na área das edificações seja necessária uma gestão para optimizar o edifício. Não é clara a utilização da palavra "lucro" no caso da Gestão de Edifícios pois a ela não está associado um retorno económico imediato. Porém, é o seu objectivo optimizar o edifício técnica, económica e funcionalmente, o que resultará num benefício económico a médio e longo prazo. A Gestão de Edifícios compreende portanto todas as acções e procedimentos necessários de efectuar a um edifício, na sua fase de utilização, para que se possa optimizar o desempenho do mesmo.

Um edifício é considerado nas vertentes de "recurso" e "bem". A noção de "recurso" é a parte física de um edifício, a construção em si, enquanto a noção de "bem" é vista como um conceito de valor, que no caso de edifícios, é um valor de longa duração. O objectivo da gestão de edifícios é portanto optimizar a performance de um edifício mantendo ou aumentando o seu valor económico.

Pode-se atingir esta performance se o gestor seguir uma linha de acção caracterizada por:

- Optimizar a utilização;
- Promover a manutenção;
- Observar comportamentos e agir em conformidade;
- Proteger.

Na gestão tradicional observa-se uma subdivisão hierárquica em Actividades, Processos e Procedimentos. É essa subdivisão que se depara a Gestão de Edifícios. Na Fig. 1 apresentam-se as três actividades da gestão de edifícios. [8]

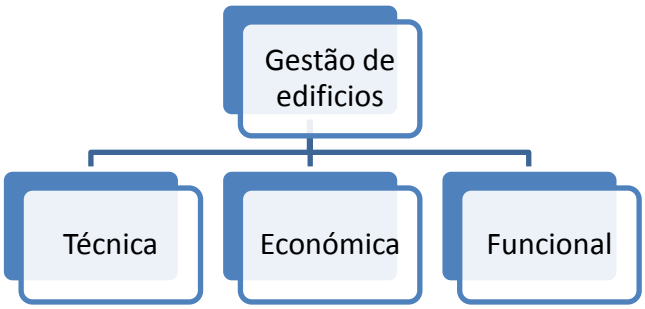


Fig. 1 - Estrutura fundamental da gestão de edifícios.

Os processos relacionados com o desempenho do edifício e de todos os seus elementos estão ligados à actividade técnica da gestão de edifícios.

Todas as despesas legais necessárias para o funcionamento do edifício para o que foi projectado estão incluídas na actividade económica da gestão de edifícios.

Na actividade funcional da gestão de edifícios estão todas as actividades necessárias para que o edifício tenha o funcionamento para o qual foi projectado.

É portanto no domínio da actividade técnica da gestão de edifícios que serão tomadas as medidas de manutenção, alvo de estudo neste trabalho.

Esses processos são por vezes enquadrados na "gestão de manutenção". Segundo a NP EN 13306:2007, gestão da manutenção define-se por "Todas as actividades de gestão que determinam os objectivos, a estratégia e as responsabilidades respeitantes à manutenção e que os implementam por diversos meios tais como o planeamento, o controlo e supervisão da manutenção e a melhoria de métodos na organização, incluindo os aspectos económicos."

3.2. MANUTENÇÃO DE EDIFÍCIOS

3.2.1. CONSIDERAÇÕES INICIAIS

Actualmente vive-se uma época de grande contenção económica, o que naturalmente se reflecte na construção. Segundo o Instituto Nacional de Estatística (INE), "O índice de novas encomendas na construção apresentou uma diminuição homóloga de 11,1% no 4º trimestre de 2010." Com estes dados é possível concluir que, com o abrandamento da construção de novos edifícios, terá que se prestar maior atenção às obras existentes de um modo mais eficiente e económico.

Durante o tempo de vida útil de um edifício, ele vai-se deteriorando gradualmente, quer devido a acções externas à própria utilização. Em consequência dessas acções os elementos e componentes do edifício vão perdendo qualidade e usabilidade para o qual foram projectados, sendo por isso necessário proceder-se a uma reparação ou substituição dos elementos afectados.

3.2.2. DEFINIÇÃO

Existem diversos trabalhos sobre manutenção de edifícios quais tratam da definição da manutenção em geral. Enunciam-se de seguida algumas dessas definições.

Assim, define-se Manutenção de edifícios como sendo "… a combinação de todas as acções técnicas e administrativas com o objectivo de reter em (...) ou, devolver a (...) elementos e componentes um estado que lhes permita desempenhar as funções para que foram projectados." [8]

Segundo a NP EN 13306:2007, manutenção é uma "Combinação de todas as acções técnicas, administrativas e de gestão, durante o ciclo de vida de um bem, destinadas a mantê-lo ou repô-lo num estado em que ele pode desempenhar a função requerida."

A NP EN 13306:2007 tem como objectivo: "…especifica termos genéricos e definições para as áreas técnica, administrativa e de gestão da manutenção. …".

Segundo a mesma norma, é importante definir os seguintes conceitos associados ao tema.

Estratégia de manutenção: "Método de gestão utilizado para atingir os objectivos de manutenção."

Plano de manutenção: "Conjunto estruturado de tarefas que compreendem as actividades, os procedimentos, os recursos e a duração necessária para executar a manutenção."

Manutibilidade: "Aptidão de um bem, sob condições de utilização definidas, para ser mantido ou restaurado, de tal modo que possa cumprir uma função requerida, quando a manutenção é realizada em condições definidas, utilizando procedimentos e recursos prescritos."

Vida útil: "Intervalo de tempo, que sob determinadas condições, começa num dado instante e termina quando a taxa de avarias se torna inaceitável ou quando o bem é considerado irreparável na sequência de uma avaria ou por outras razões pertinentes."

A manutenção é, portanto, toda a acção efectuada no edifício para que este continue a servir de acordo com o que foi projectado, com a mesma ou maior qualidade e sem perigo para os utilizadores e mantendo a mesma performance. Isto cobre tanto acções ao nível da própria construção como também todas as despesas legais para manter o edifício em funcionamento.

3.2.3. Tipos e estratégias de manutenção

Existem várias possibilidades de executar a manutenção a um Elemento Fonte de Manutenção de um edifício. Dando um exemplo, tanto se pode esperar que um elemento fonte de manutenção, demonstre clara degradação para ser reparado, como se pode reparar antes mesmo dessa demonstração, tendo por base estudos ou previsões. Abordam-se os diferentes tipos de manutenção, dando alguns exemplos práticos se necessário.

Segundo a NP EN 13306:2007 têm-se portanto os seguintes tipos e estratégias de manutenção:

- Manutenção preventiva
- Manutenção programada
- Manutenção sistemática
- Manutenção condicionada
- Manutenção preditiva
- Manutenção correctiva
- Manutenção remota
- Manutenção diferida
- Manutenção de urgência

As definições das mesmas serão transcritas da NP EN 13306:2007, devidamente identificadas.

3.2.3.1. Manutenção preventiva

"Preventivo – adj. Que tem por fim acautelar ou impedir" (Dicionário Priberam da Língua Portuguesa).

Como o nome indica, com esta manutenção tenta-se impedir que o elemento chegue a um ponto de degradação de funcionamento ou que não seja utilizável.

"Manutenção efectuada a intervalos de tempo pré-determinados, ou de acordo com critérios prescritos, com a finalidade de reduzir a probabilidade de avaria ou de degradação do funcionamento de um bem."

3.2.3.2. Manutenção programada

"Manutenção preventiva efectuada de acordo com um calendário pré-estabelecido ou de acordo com um número definido de unidades de utilização." [NP EN 13306:2007]

3.2.3.3. Manutenção sistemática

"Manutenção preventiva efectuada a intervalos de tempo preestabelecidos ou segundo um número definido de unidades de utilização mas sem controlo prévio do estado do bem." [NP EN 13306:2007].

3.2.3.4. Manutenção condicionada

"Manutenção preventiva baseada na vigilância do funcionamento do bem e/ou parâmetros significativos desse funcionamento, integrando as acções daí decorrentes." [NP EN 13306:2007].

3.2.3.5. Manutenção preditiva

"Manutenção condicionada efectuada de acordo com as previsões extrapoladas da análise e da avaliação de parâmetros significativos da degradação do bem." [NP EN 13306:2007].

3.2.3.6. Manutenção correctiva

"Manutenção efectuada depois da detecção de uma avaria e destinada a repor um bem num estado em que pode realizar uma função requerida." [NP EN 13306:2007].

3.2.3.7. Manutenção diferida

"Manutenção correctiva que não é efectuada imediatamente depois da detecção de um estado de falha, mas que é retardada de acordo com as regras de manutenção determinadas." [NP EN 13306:2007].

3.2.3.8. Manutenção de urgência

"Manutenção correctiva que é efectuada imediatamente após a detecção de um estado de falha, para evitar consequências inaceitáveis." [NP EN 13306:2007].

3.3. DADOS DE MANUTENÇÃO A IMPLEMENTAR NA BASE DE DADOS

Existem várias informações ligadas a Elementos Fontes de Manutenção importantes para uma boa manutenção do mesmo.

Um edifício é um conjunto de elementos ou componentes que, seguindo o projecto, se materializam sistemas construtivos. Nesse conjunto de elementos, todos estão sujeitos a uma deterioração diferente e, por isso, na análise de cada elemento ao nível da manutenção, destinam-se acções de manutenção diferentes pois cada um tem o seu comportamento. Por esses motivos, todos eles são elementos que necessitam de manutenção e as intervenções feitas são diferentes para todos eles.

A informação básica, que estará presente em praticamente todos os elementos, é a empresa fornecedora do material, e as garantias associadas a eles. Dentro da informação da empresa importa referir que é impreterível ter acesso aos contactos da empresa, condições de montagem, empresa que montou, no caso de a empresa fornecedora e montadora serem diferentes.

Há também que salientar que é importante o acesso a informações tais como o projecto inicial, *as build*, data de instalação, entre outras. Todos estes dados são fixos, que poderão constar nos livros de obra e do projecto inicial da obra.

A partir do momento de utilização do edifício, é fundamental ter acesso a dados como alterações posteriores, manutenções efectuadas, observações e todo um registo que contenha as alterações efectuadas num dado elemento. Esta informação é útil para que não se percam nem caiam no esquecimento alterações ou manutenções a certos elementos. Com base nestes dados propõe-se uma estrutura do que podem ser as informações presentes numa base de dados para apoio à manutenção, ainda que de um modo bastante superficial.

Componentes de projecto:

- Fabricante;
- Fornecedor;
- Equipa de montagem;
- Garantias;
- Elementos de projecto (desenhos simples ou ficheiros CAD).

Manutenção:

- Manual de inspecções;
- Manual de limpeza;
- Substituições;
- Manual de utilização;
- Emergências.

Comportamento:

- Registo de ocorrências.

4
RFID

Neste Capítulo analisam-se as características da tecnologia RFID. Esta análise não visa ser uma caracterização completa da tecnologia, pois os conceitos associados não se encontram no âmbito deste trabalho.

4.1. CONTEXTUALIZAÇÃO HISTÓRICA

Ao longo do tempo e com o desenvolvimento da indústria, sentiu-se a necessidade de criar sistemas automáticos de identificação.

A solução mais conhecida e mais enraizada no mercado actual é o Código de barras. O sistema de código de barras começou por ser um trabalho de *Bernard Silver*, um estudante de *Philadelphia's Drexel Institute of Technology*, em 1948, que usava um padrão com tinta ultravioleta. Esse sistema não foi adoptado pela indústria por causa do seu preço e da instabilidade da tinta. [9]

Entre 1949 e 1973 muitos foram os desenvolvimentos, destacando-se *Norman Joseph Woodland*, que trabalhando na IBM, conseguiu desenvolver o que hoje se encontra em todos os supermercados, embalagens, partes de automóveis e nos mais variados produtos, que em certa altura da produção, ou já na própria venda necessitem de serem identificados, o Código de barras. [10]

Fig. 2 - Exemplo de um código de barras.

Existe também um novo padrão para código de barras, QR Code, que já consegue armazenar mais informação que o tradicional código de barras, e está gradualmente a espalhar-se o seu uso devido à grande informação e à facilidade de leitura, que pode ser feita por qualquer aparelho que possua uma máquina fotográfica, tal como o telemóvel. Apesar desta evolução, este sistema tem a obrigatoriedade de estar no campo de visão do leitor. [11]

Fig. 3 - Exemplo de um QR Code com a informação em texto do título deste trabalho.

Nos últimos tempos, tem-se vindo a desenvolver cada vez mais a tecnologia RFID, sigla que significa *Radio-Frequency Identification*, que cuja tradução corrente corresponde a Identificação por Radiofrequência. Apesar da sua intenção ser idêntica à do código de barras, tem inúmeras vantagens sobre este, embora apresente algumas desvantagens.

O enquadramento histórico da tecnologia RFID foi baseada em [12], [13] e [14].

O sistema RFID foi desenvolvido durante a Segunda Guerra Mundial. Nessa altura já existiam os radares que detectavam aviões, mas não conseguiam distinguir um avião aliado de um inimigo. É então que *Sir Robert Alexander Watson-Watt*, graduado com um bacharelato em Engenharia em 1912, também ele inventor do radar, começou a estudar uma forma de conseguir identificar a origem dos aviões que surgiam no radar.

Esse sistema, apelidado de IFF, sigla de *Identify Friend or Foe* (Identificador de Amigo ou Inimigo), consistia em se colocar um transmissor em cada avião aliado. Então, quando esses transmissores recebiam as ondas dos radares, respondiam com outro sinal, dando conhecimento que era aliado.

A tecnologia RFID funciona nos mesmos parâmetros: um sinal é enviado a um *transponder*[2] (ou *tag*), o qual é activado e reflecte de volta o sinal (sistema passivo) ou transmite o seu próprio sinal (sistemas activos).

Vários estudos e aplicações foram efectuados nesta tecnologia antes da aplicação e do modo como hoje em dia é vista a RFID.

Em 1973, *Mario W. Cardullo* registou a primeira patente, de um *tag* RFID passivo com memória regravável. Acredita-se que este é o antecessor da tecnologia RFID moderna. Nesse mesmo ano foi registada a outra patente RFID, por *Charles Walton*, mas desta vez para um sistema que utiliza um *tag* passivo e um leitor para activar um mecanismo de abertura de uma porta.

O Laboratório Nacional de *Los Alamos*, Estados Unidos da América (EUA), teve um papel preponderante no desenvolvimento desta tecnologia. A pedido do Departamento de Energia dos EUA, este laboratório desenvolveu um sistema para monitorizar/rastrear material nuclear que se baseava em colocar leitores RFID nos portões das instalações e *tags* nos camiões de transportes, podendo assim trocar informação de identificação do material ou outras informações importantes (nome do condutor, etc.). Este mesmo método começou a ser comercializado por volta de 1980 para o pagamento de portagens em auto-estradas, pontes e túneis em todo o mundo. O mesmo laboratório, a pedido do Departamento da Agricultura dos EUA, desenvolveu um sistema para poder monitorizar animais nas

[2] *Transponder* é uma abreviação de *transmitter-responder*. O seu funcionamento consiste em receber uma onda rádio, e como o próprio nome diz, responder a essa com outra onda rádio.[15]

quintas. Isto porque nas grandes quintas era difícil saber se determinado animal já tinha ou não sido medicado, levando à morte de alguns animais por sobredosagem. Desenvolveu-se então um *tag* RFID UHF (frequência muito alta ou *Ultra High Frequency*), o qual transmite ondas que têm uma frequência entre 300MHz e 3 GHz. O *tag* RFID UHF foi colocado exteriormente no animal, recebendo a energia das ondas do leitor e reflectindo um sinal com uma identificação, sendo possível então verificar se esse animal já tinha tomado a dose certa de medicação. Mais tarde foi desenvolvido um sistema idêntico mas a operar em baixa frequência o que permitiu injectar *tags* passivos encapsulados em vidro, para o interior de animais.

Com ondas UHF, as informações entre leitor e *tag* são transferidas a grandes velocidades, mas têm a desvantagem de não conseguir atravessar materiais com grande percentagem de água.

Esta tecnologia é hoje usada em Portugal para controlar e cobrar portagens nas antigas *SCUT*.

Hoje em dia, 13,56 MHz é a frequência usada para controlo de acessos, sistemas de pagamento e cartões inteligentes sem contacto. Há também um sistema anti-roubo usado em alguns automóveis, que consiste numa antena perto da ignição, que espera pela resposta do *tag* embutido no comando do carro para desbloquear a ignição do mesmo e que caso não tenha a resposta esperada não deixa ligar o carro.

No início dos anos 90, a IBM (*International Business Machines*) desenvolveu e patenteou um sistema RFID de muito alta frequência que oferecia uma rápida transferência de dados e apresentava uma distância de leitura de cerca de 20 metros em boas condições. Foram feitos testes em algumas redes comerciais como o *Wall-Mart*. Devido às más condições financeiras da IBM, a patente foi vendida a meio dos anos 90 à empresa *Intermec*, empresa especializada em sistemas de código de barras.

O custo associado a esta tecnologia foi um impasse para a sua evolução, pois as empresas só adquiriam esta tecnologia quando tivesse um custo de produção mais baixo, e os custos de produção só diminuiriam quando a tecnologia fosse fortemente adoptada.

Só a partir de 1999 se começou a registar uma expansão da tecnologia, quando o custo associado a esta tecnologia deixou de ser um entrave para a propagação da mesma e quando a *Uniform Code Council*, *EAN International*, *Procter & Gamble* e a *Gillette* fundaram a *Auto-ID Center* no Instituto de Tecnologia de *Massachusetts*. *David Brock* e Sanjay Sarma investigaram a possibilidade de colocarem um sistema RFID de baixo custo em todos os produtos para poderem localizá-los no mercado. Estes *tags* seriam passivos e, apenas com um número único de memória, seriam mais baratos de adoptar do que *tags* com memória interna e portanto com um chip mais complexo. Os dados associados a cada número único estariam acessíveis através da internet. Estes dois professores mudaram a maneira de pensar no tema de RFID em produtos pois, antes do estudo deles, os dados relativos ao número único estariam numa base de dados móvel. A partir deste momento, é então possível rastrear um produto desde o momento que saiu da fábrica, nos armazéns, nas empresas de transporte até chegar ao comprador final.

Entre 1999 e 2003, mais de 100 grandes empresas de venda ao público começaram a apoiar o *Auto-ID Center*, incluindo o Departamento de Defesa dos EUA e muitos vendedores da tecnologia RFID. Foram abertos laboratórios de investigação na Austrália, Reino Unido, Suíça, Japão e China. Foram desenvolvidas várias normalizações, como dois protocolos de interface aéreo (Class 1 e Class 0), esquema de numeração do número electrónico de produto (*Electornic Product Code* – EPC), assim como a arquitectura de rede para localizar informação associada a um *tag* RFID na internet. Esta tecnologia foi certificada pelo *Uniform Code Council* em 2003 e foi também criada a *EPCglobal*, que

é uma organização sem fins lucrativos cuja função é administrar e fomentar o desenvolvimento da tecnologia, criando uma padronização para as diferentes aplicações.

O *Auto-ID Center* é uma colaboração entre empresas privadas e o mundo académico com fins não lucrativos que desenvolveu uma estrutura para rastrear produtos globalmente. Fechou portas em Setembro de 2003, mas os laboratórios que abriu continuam o trabalho de investigação.

Em Dezembro de 2004, foi rectificado o protocolo EPC, para o EPC Gen 2. Ele foi desenhado para uso internacional e possui outras melhorias tais como o modo de leitura "densa", ou seja, impedir colisões entre leitores quando muitos coexistem na mesma área.

Hoje em dia esta tecnologia tem inúmeras aplicações. É utilizada em Hospitais para monitorizar crianças recém-nascidas, assim como localizar equipamentos médicos. É utilizada em veículos para pagamentos de portagens, parques etc. A nível industrial tem várias aplicações, entre elas na gestão de stock, localizações de produtos quer na linha de montagem quer no transporte, controlo de pessoal etc. Na área comercial utiliza-se este sistema na segurança de lojas em que os produtos ao sair da loja emite um aviso caso não tenha sido pago. Na área da manutenção o Aeroporto de Frankfurt iniciou um projecto-piloto em 2003 com o objectivo de testar os benefícios do RFID na manutenção dos seus 22 mil extintores. Neste projecto concluiu-se que a utilização deste sistema permite um ganho extraordinário de eficiência, fazendo com que tal processo reduzisse a utilização de 88 mil páginas de papel arquivadas por ano.

Conclui-se que esta tecnologia está actualmente em grande desenvolvimento e que um dos seus maiores entraves sejam a adopção em massa do sistema por parte das possíveis aplicações.

4.2. CONSIDERAÇÕES INICIAIS

RFID é uma sigla de *Radio-Frequency IDentification*. É um termo genérico usado para descrever um sistema que transmite a identidade via ondas de rádio. É uma tecnologia de identificação automática, assim como o código de barras.

Um sistema RFID é constituído por dois componentes de *hardware*, um conjunto leitor antena e um transponder (*tag* ou etiqueta).

O objectivo é que o leitor consiga ler a informação guardada no *tag* e transmiti-la para um computador.

O seu funcionamento começa com o leitor a enviar um sinal através de uma radiofrequência. O *tag*, sintonizado para receber esse comprimento de onda, transmite a informação armazenada também através de radiofrequência, o leitor recebe-o, descodifica-o e envia-o para um computador.

Geralmente, dentro do leitor, existe um *middleware*, cuja função principal é controlar o funcionamento da antena e fazer um tratamento de dados antes de ser enviado o resultado final para o computador. Existem também casos de o *middleware* estar localizado num servidor central, mas em situações mais complexas. [13]

Após a recepção da "resposta" do *tag* e, com o devido tratamento do *middleware*, é necessário ter um software capaz de interpretar os dados recebidos.

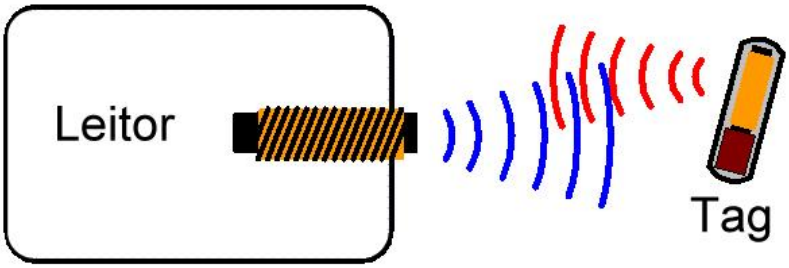


Fig. 4 - Esquema do funcionamento de um sistema RFID.

4.3. TAG

O *Tag* RFID é um *transponder* que pode assumir os mais variados formatos, assim como várias especificações técnicas, dando assim uma grande margem para a escolha acertada num determinado caso.

Antes de mais, o seu formato pode assumir a forma de cartões, porta-chaves, pulseiras, autocolantes, etiquetas, grandes caixas e até objectos do tamanho de um grão de arroz.

Para além da variação de formato, existe também uma grande hipótese de escolha técnica. Há *tag*s passivos, activos, semi-activos e semi-passivos. Estes são dispositivos que dependem do tipo de alimentação eléctrica para o envio da sua onda. [16]

Os *tags* podem funcionar nas mais variadas frequências, o que também é tido em conta na escolha para um dado problema, pois influencia o seu preço, os materiais que atravessam, a rapidez de comunicação assim como a distância máxima de leitura ou escrita.

Um *tag* pode receber energia eléctrica através do campo indutivo gerado pelo leitor ou pode possuir uma bateria interna que utiliza para enviar as ondas.

Chamam-se *tags* passivos aqueles que aproveitam a energia do leitor e *tags* activos aos que possuem uma bateria interna.

Além disto, os *tags* podem possuir diferentes tipos de memória: memória apenas de leitura (ROM-*Read Only Memory*) como o seu código único de identificação, memória WORM (*Write Once Read Many*) que possibilita apenas uma escrita para posterior leituras, ou memórias que possibilitam ao utilizador guardar e alterar dados.

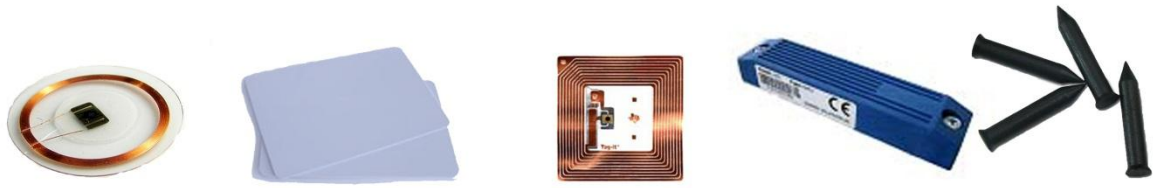


Fig. 5 - Vários formatos de tags, respectivamente: Moeda, Cartão plástico, Autocolante, Tag para ferro e Tag em forma de prego.

4.3.1. *TAGS* PASSIVOS

Estes *tags* caracterizam-se por não necessitarem de ter uma bateria interna para responder a uma onda enviada pelo leitor, fazendo uso do campo induzido enviado pelo leitor para produzir energia suficiente para enviar a sua onda de resposta. São os mais baratos do mercado e talvez por isso mesmo os mais adoptados. Podem funcionar em baixa frequência, alta frequência e muito alta frequência. Os de baixa frequência operam entre os valores de 124, 125 ou 135 kHz, os de alta frequência operam a 13,56Mhz e os de muito alta frequência operam em qualquer banda entre 860 a 960 MHz.

A diferentes frequências de operação correspondem diferentes propriedades e claro, diferentes aplicações. No subcapítulo 4.6, estas noções são clarificadas com maior detalhe.

A distância máxima de leitura de um *tag* passivo é menor do que um activo, geralmente compreendida entre alguns centímetros até perto de 10 metros. Isto porque o sinal reflectido pelo *tag* tem apenas uma fracção da energia recebida do leitor, dependendo assim a sua distância de leitura da potência da antena do leitor.

4.3.2. *TAGS* ACTIVOS

Ao contrário dos *tags* passivos, que utilizam a energia induzida pelo leitor, estes têm uma bateria incorporada que permite alimentar electricamente o envio da onda de resposta.

Por possuir uma bateria para alimentar o envio dessas ondas, a sua distância de leitura é bastante superior a uma passiva, podendo alcançar os 100 metros.

São geralmente utilizados em grandes objectos, tais como contentores de carga, e funcionam em frequências altas, normalmente 455MHz, 2.45GHz ou 5.8GHz (microondas).

4.3.3. *TAGS* SEMI-PASSIVOS

Um *tag* semi-passivo é um *tag* passivo que utiliza uma bateria para alimentar o seu circuito interno mas não para alimentar o sinal de resposta. Se essa energia é utilizada para accionar um sensor, são denominados por *tags* de sensor. São mais pequenos que os *tags* activos e possuem mais funcionalidades que os *tags* passivos pois possuem energia própria.

4.3.4. TAGS SEMI-ACTIVOS

Um *tag* semi-activo é idêntico a um *tag* activo mas permanece inactivo até receber um sinal vindo do leitor. Tal como os *tags* activos, podem comunicar a grandes distâncias. A maior diferença entre eles é que um *tag* semi-activo possui uma bateria que habitualmente tem maior duração pois não está sempre activo, mas tem como desvantagem o tempo de resposta, pois, por se encontrar em modo de espera, demora mais tempo a responder ao sinal do leitor.

4.4. LEITORES/ANTENAS

A função geral do leitor é aceder à informação contida no *tag*. Este envia uma onda, em que o *tag* recebe e responde com outra onda, na frequência que o leitor está em modo de escuta. Geralmente, um leitor possui duas antenas: uma para enviar uma frequência e outra para registar a frequência vinda do *tag*. Depois de receber a onda do *tag* e de ser tratado pelo *middleware*, o leitor envia a informação para um computador/software para que seja lido pelo utilizador.

A antena tanto se pode encontrar no interior do leitor, como no seu exterior, estando ligada ao leitor por uma porta. Além disso, é possível também que o leitor possua várias antenas.

Um leitor pode estar configurado para funcionar só num protocolo, baseado nas normas ISO ou nas normas EPC, ou, pela força do mercado, pode ser multiprotocolar, podendo ler *tag*s vindos de fábricas onde não se use o mesmo protocolo e ser usado em países de diferentes protocolos.

Um leitor pode ter uma ligação ao computador por várias interfaces: USB, *Ethernet*, comunicação série, *WiFi*, *Bluetooth* e outras. É pela porta de ligação que o leitor envia a informação tratada pelo middleware para o computador para o software ler os dados. De notar que um leitor *WiFi* ou *Bluetooth* não necessita de estar fisicamente ligado ao computador em que corre o software.

O leitor pode criar dois tipos de campos de onda: um linear e um circular.

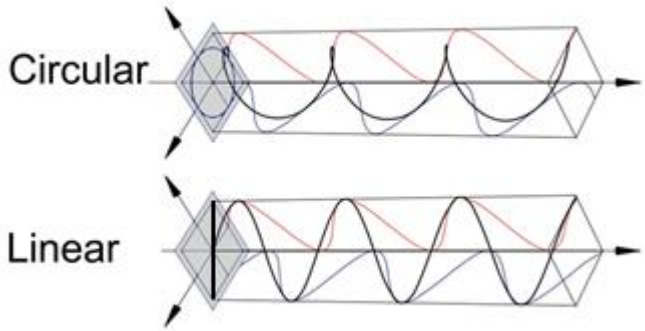


Fig. 6 - Exemplo de uma Polarização Circular e Linear.

Um campo linear tem mais capacidade de penetração e consegue efectuar uma leitura mais distante do que se fosse um campo circular, mas em contrapartida com recurso a um campo circular a probabilidade de localizar o *tag* é maior, devido à não necessidade de apontar directamente para este. As ondas de um campo linear podem ser horizontais ou verticais. [17]

4.5. SOFTWARE

O software RFID é o que desempenha a função de "humanizar" a leitura dos dados do leitor. No caso deste trabalho, o software tem que ler o número único do RFID e localizar toda a informação numa base de dados remota ou descodificar a memória interna do *tag* para uma visualização desses dados.

4.6. FREQUÊNCIAS DE OPERAÇÃO RFID

A frequência com que um sistema RFID opera tem influência não só na velocidade de leitura, mas também na distância máxima de leitura e materiais que são possíveis atravessar.

Baixa Frequência (LF – *Low Frequency*) : Esta gama é caracterizada por se encontrar entre 9kHz e 135kHz. Os sistemas que utilizam esta gama têm uma distância de leitura pequena (apenas de alguns centímetros) e têm a capacidade de penetrar objectos com grandes quantidades de água, tal como animais e mesmo seres humanos.

Alta Frequência (HF – *High Frequency*) : 13,56MHz é a frequência utilizada nesta gama. Caracteriza-se por distâncias de leitura de um centímetro a um metro e meio. Geralmente os *tag*s desta gama são passivos.

Muito Alta Frequência (UHF – *Ultra High Frequen*cy) : Encontra-se na gama entre 300MHz e 1,2GHz. Não conseguem atravessar metal nem objectos constituídos por muita água, mas conseguem obter velocidades de transmissão elevadas. Nos Estado Unidos da América, a gama encontra-se entre 902 e 928MHz enquanto, na Europa, a gama situa-se entre 865.7 e 867.5 MHz.

Microondas : Encontram-se entre 2.45GHz e 5.8GHz e têm como vantagem a resistência a fortes campos electromagnéticos. São utilizados em linhas de montagem que possuam grandes motores eléctricos e sistemas de soldadura, assim como em situações onde é necessário uma grande distância de leitura. Estes *tags* necessitam de mais energia e são sistemas mais caros.

4.7. REGULAMENTOS E NORMAS

Devido a este sistema ser desenvolvido por diversas empresas em várias frequências de operação, deu-se a necessidade da criação de protocolos e normas para que, de um certo modo, as várias soluções fossem compatíveis entre si. Foi uma tarefa dificultada dado não existir um consenso entre os vários fabricantes de sistemas RFID. Para o funcionamento globalizado destes sistemas foi necessária a criação de protocolos de comunicação relacionados com as frequências de operação, onde foram criadas pela Organização Internacional de Normalização as seguintes normas[19]:

- 18000–1: Parâmetros genéricos para interface aérea de frequências globalmente aceites.
- 18000–2: Interface aéreo para 135 KHz.
- 18000–3: Interface aéreo para 13.56 MHz.
- 18000–4: Interface aéreo para 2.45 GHz.
- 18000–5: Interface aéreo para 5.8 GHz.
- 18000–6: Interface aéreo para 860 MHz até 930 MHz.
- 18000–7: Interface aéreo de 433.92 MHz.

Existem também normas ISO que regulam algumas das aplicações desta tecnologia, como é o caso da ISO 14233, em que se trata da identificação animal por RFID.

Mais tarde foi criado pela EPCglobal a norma UHF EPC Gen 2. Esta norma foi criada para facilitar o uso do Código Electrónico de Produto (EPC – *Electronic Product Code*) que permite a identificação de objectos únicos[20]. Uma estrutura de rede EPC caracteriza-se por essencialmente cinco elementos:

- Número EPC: Identificador global e único, que serve para realizar consultas sobre o objecto que ele identifica.
- Etiqueta EPC (tag): Portadora de dados (EPC) que comunica com o leitor por RF. É constituída por memória, microprocessador e por uma antena.
- Reader/Leitor: Dispositivo de captura de dados; portátil ou fixo, que se conecta à rede
- EPC. Savant (EPC Middleware): Software que controla os leitores. Pode funcionar com um repositório local de números EPC's e informações associadas.
- ONS (Object Name Service): recurso partilhado que possuí informações associadas ao número EPC (equivalente ao DNS para internet).
- EPCIS (EPC Information Service): Serviço de informações de EPC's que contém todos os dados relativos a um EPC. (Utiliza PML que é uma linguagem definida em XML, para permitir consultas e obter dados relacionados com os números EPCs).

5
PROPRIEDADES DO SISTEMA RFID

Neste capítulo analisam-se as propriedades de um sistema RFID para aplicação à manutenção de edifícios. Em princípio antevê-se um cenário ideal, que em correlação com a realidade tecnológica e uma análise financeira se conclui um sistema possível para aplicação real.

5.1. CENÁRIO IDEAL

Expondo um cenário que o autor considera óptimo, faz-se uma abordagem aos custos e à tecnologia existente.

O cenário óptimo consiste em que todos os elementos passíveis de manutenção de um edifício contenham um sistema de identificação automático que permitisse ser lido á distância. Isto aplicar-se-ia a elementos pré-fabricados de estrutura, assim como a todos os elementos estruturais executados na obra.

Estes elementos construtivos podem também ter monitorização de informações relevantes ao desempenho do elemento ou à necessidade de manutenção, tais como humidades, reconhecimento de fendas, temperaturas, níveis de utilização, etc.

Elementos não estruturais, tais como portas, sistemas eléctricos e mecânicos, instalações sanitárias assim como sistemas AVAC (Aquecimento, Ventilação e Ar Condicionado), teriam também sistemas de identificação à distância, assim como sistemas de monitorização de propriedades que interferem na manutenção e desempenho dos mesmos, tais como humidade nos sistemas sanitários, vibrações em sistemas mecânicos, níveis de utilização para portas, etc.

Em todos os elementos, além da sua identificação electrónica à distância, devem também ter associado a eles uma série de informações fixas, tais como fabricante, equipa de montagem, garantias, manuais de manutenção, com informações sobre limpezas, inspecções, pro-acções, substituições, correcções e outras relevantes ao assunto.

Além da informação fixa, teriam também de ter a propriedade de poder gravar nesse sistema um registo/cadastro de todas as acções efectuadas com informações de datas, estados, identificação de quem efectuou essas acções, observações, etc.

Com a informação da monitorização das propriedades referentes ao desempenho e à manutenção, os elementos têm a capacidade de transmitir um aviso caso algum dos valores dessas propriedades atinja um nível que necessite de intervenção.

Da mesma maneira, pode também obter-se alertas quando algum elemento móvel atingir um nível de utilização que necessite de manutenção.

Este seria um cenário óptimo que não tem em consideração quer a tecnologia aplicada quer os custos envolvidos. Proceder-se-á então de forma forma mais objectiva.

5.2. ASSOCIAÇÃO DO CENÁRIO À REALIDADE TECNOLÓGICA

Existem várias tecnologias de identificação, entre eles, os já identificados, código de barras, QR Code, RFID. Para esta aplicação, o autor sente a necessidade de identificar os EFM com um sistema que não implique visão directa para o identificador. Logo, a tecnologia RFID será aplicada neste trabalho, pois é a única que, graças ao uso de radiofrequência para comunicação, possui esta propriedade.

Sabe-se já que os *tag*s RFID têm, hoje em dia, potencialidades para permitir que o cenário considerado ideal se torne realidade.

Existem *tags* com memória interna, com a propriedade de leitura/escrita, onde daria para guardar toda a informação fixa descrita, como também adicionar uma ficha registo sempre que existir uma manutenção a esse elemento, tendo como limite a memória do próprio *tag*.

Os *tags* podem ser lidos a grandes distâncias, dependendo da sua frequência. Geralmente, e por observação dos produtos disponíveis no mercado, quanto maior a frequência, maior é a distância a que se podem realizar operações de leitura/escrita, não esquecendo que diferentes frequências atravessam diferentes materiais.

Os *tags* podem também ter um elevado tempo de vida, com empresas a afirmarem que um tag passivo, desde que bem conservado, pode ter um tempo de vida praticamente ilimitado, o que é bastante relevante para este projecto.

5.3. ANÁLISE ECONÓMICA

Os *tags* activos são os que mais memória possuem e os que teriam uma distância de leitura/escrita superior, mas, devido ao seu alto custo e ao uso de baterias que diminuiriam significativamente o seu tempo de vida, não se adaptam às características do projecto.

Consequentemente, todas as informações de monitorização que poderiam vir de um *tag* equipado com sensores, estão também descartadas para elementos fixos, pois estes também precisariam de uma bateria ou mesmo de estarem ligadas à corrente eléctrica.

Para elementos fixos de construção, os *tags* activos não seriam os ideais, mas poderiam ser pensados para a utilização em elementos mecânicos e eléctricos, tais como portas automáticas, elevadores, AVAC, o que seria um caso objecto de estudo mais aprofundado sobre relação custo/benefício e que estaria directamente ligado às características da obra específica.

Ir-se-ão então utilizar *tags* passivos, os quais, além de poderem ter uma memória interna, podem ser lidos a distâncias suficientes para o projecto (aproximadamente dez metros), têm um tempo de vida útil superior a vinte anos e são de custo reduzido (0,16€ a 0,20€ por unidade).

5.4. CONDIÇÕES LIMITE DE LOCALIZAÇÃO DOS TAGS

Analisam-se agora as condições limites das diferentes maneiras de implementar o sistema, verificando a quantidade e localização dos *tags* que se utilizarão.

Por um lado podemos ter um edifício com todos os seus elementos identificados com *tags*, podendo até dividir um elemento inteiro, por exemplo porta, por vários componentes que compõem o elemento, que poderiam ser fechadura, dobradiças, madeira, etc. Por outro lado, poderemos identificar um edifício apenas com um *tag*.

5.4.1. Condição menor

A condição limite que permite o menor número possível de *tags* poderia ser um *tag* por obra, colocado numa zona do edifício. Essa localização teria de ser normalizada. Uma hipótese seria a sua colocação na porta de entrada do edifício ou, por segurança, dentro de uma parede da entrada.

Com esta utilização, perder-se-ia a vantagem de apontar para um elemento e obter as informações directamente, o que só poderia ser executado através de um software que, depois de ler o *tag*, permitisse navegar por todos os elementos existentes nesse edifício.

Economicamente o sistema seria mais económico, pois utilizaria um número bastante inferior de *tags*, que apesar de não terem um custo significativo na construção, representam um factor a ter em conta devido à sua implementação.

É uma solução que se apresenta tentadora, quando se implementa este sistema num edifício já construído. O seu custo de implementação seria bastante mais reduzido. Por outro lado, aplicar um *tag* dentro de cada elemento de um edifício existente é praticamente inviável.

5.4.2. Condição maior

Nesta solução, todos os elementos fontes de manutenção teriam o sistema de identificação. Como já se referiu, não só se identificariam todos os elementos, como também os componentes de que é constituído, podendo deste modo estarem, por exemplo, o elemento "porta" e o componente "dobradiça" identificados, sendo que o elemento "porta" seria um agrupamento dos componentes que a constitui.

Cada elemento poderia ter vários *tags* para o identificar. Por exemplo, uma parede com 25 metros de comprimento poderia ter *tags* de 5 em 5 metros e ser assim identificada em qualquer local dessa parede.

Com esta utilização ganhar-se-ia o facto de ter toda a informação necessária directamente no elemento, sem necessidade de um software para navegar pelos elementos do edifício.

Um dos problemas que o autor antevê seria o excesso de informação, pois uma antena RFID com longo alcance iria localizar demasiados componentes, o que poderia conduzir a desinformação ou mesmo confusão das pessoas responsáveis pela manutenção.

5.4.3. Condição intermédia

Nesta solução apenas se identificam os elementos fontes de manutenção como um todo, podendo dentro da informação contida no *tag* referenciar todos os componentes que nela estão contidos. Por exemplo, numa porta poderemos saber quem foi o fabricante da porta, mas também podemos saber o fabricante e a referência da fechadura, dobradiças, etc. sem necessidade de os identificar com um *tag*.

No caso de grandes componentes, não se usariam vários *tags* para podê-lo identificar em qualquer sítio, mas sim um sítio normalizado para cada componente. Por exemplo, uma parede terá o seu *tag*

identificador no canto inferior esquerdo, o que geraria uma discussão sobre qual a localização mais proveitosa para cada componente.

Imaginemos o caso de uma pequena sala de arrumos. Neste caso, em vez de ter um *tag* em cada parede, na porta, no interruptor, candeeiro, etc., poderíamos ter apenas um *tag* na entrada da sala e por software localizar todos os elementos presentes nessa sala.

5.4.4. CONCLUSÃO

Concluímos que, cada condição pode ser utilizada, apenas dependendo da característica da obra.

Para um edifício já existente, a condição de utilizar o menor número de *tags* possíveis, que tanto poderia passar por ter apenas um por edifício ou mesmo um por andar, seria a melhor hipótese. Adoptar outras soluções que utilizariam mais *tags* seria uma intervenção bastante morosa e com grandes custos associados.

Para um edifício em fase de projecto/construção, poder-se-ia adoptar tanto a condição máxima como a condição intermédia. A escolha de uma delas dependeria no factor de exigência da manutenção do edifício.

5.5. MEMÓRIA INTERNA E/OU BASE DE DADOS

Como já foi visto, os *tags* podem possuir memória interna. Um dos problemas que se prevê para a utilização da memória interna do *tag* para armazenar informação será a necessidade de normalização da introdução dessa informação para que o software consiga entender a informação contida no *tag*. Admitindo que as mais variadas empresas fabricantes poderiam montar componentes já com *tags* embutidos seria conveniente que todas elas respeitassem o mesmo padrão no tratamento dessa informação.

Além disto, estando a informação na memória do *tag* do elemento, no caso de avaria do *tag*, perder-se-ia toda a informação nele contido. Este problema seria facilmente contornado, se, além da informação estar contida na memória interna do *tag*, estivesse também armazenada numa base de dados informática, actualizando-se bilateralmente.

Para resolução deste problema e, de certo modo não entrar no campo de normalizações, iremos utilizar um sistema, que apenas faz uma leitura do *tag*, obtendo o numero único de identificação, que referenciará a um elemento numa base de dados, onde estará toda a informação necessária e permitindo adicionar tanta informação quanto o utilizador final necessitar.

5.6. LEITORES E TAGS

Depois de vários contactos com as maiores empresas fabricantes e de desenvolvimento desta tecnologia, obteve-se um leque de opções no que diz respeito aos leitores e antenas.

Existe uma variedade de opções que variam desde tamanho, peso, conectividade, alimentação, distância de leitura, frequência, polarização da onda, etc.

Apresentadas as características principais do projecto aos vários fabricantes, obtivemos respostas iguais para as frequências de funcionamento: devem ser utilizadas frequências UHF.

Existem leitores que possuem uma antena integrada e leitores em que se podem ligar uma ou mais antenas, dependendo da necessidade. Os mais adequados a este projecto serão os que possuem a antena integrada por uma questão de mobilidade.

Começando pela conectividade, temos conjuntos de leitor e antena, que comunicam com o computador através de ligações *USB*, *Ethernet*, *Wi-Fi*, *Bluetooth*, *RS 232*, *PC Card2*, entre outros. Quanto à ligação, não haverá qualquer tipo de preferência, pois todos eles funcionam de maneira perfeita para o projecto em questão.

Salienta-se que as ligações por *Wi-Fi* e *Bluetooth* não necessitam de cabos para ligar ao computador ou *tablet*. Este facto não traria grandes vantagens pois, como toda a informação apareceria no computador, o conjunto leitor/computador teriam de andar lado a lado. Também se teria a desvantagem de estes leitores possuírem uma bateria que necessariamente teria de ser carregada, para além da bateria do próprio computador portátil ou *tablet*.

Os leitores por ligação *USB*, *Ethernet* e *PC Card* têm a capacidade de serem alimentados através da própria ligação ao computador.

Todas as ligações têm diferentes velocidades de transmissão de dados, mas, para o projecto em questão, as diferenças de velocidades não são relevantes pois esta é uma tecnologia que permite centenas de leituras por segundo.

No entanto, existem no mercado vários modelos que têm o leitor, antena e um computador PDA num só conjunto. Esta solução é a que mais se adequa ao projecto, pois consiste em apenas um aparelho que possui o necessário para a tarefa requerida. Além disso, são aparelhos cuja montagem é pensada para resistir a ambientes mais exigentes.

Apresenta-se agora uma solução possível para conjunto leitor, antena e computador. É fabricado pela *Motorola* e tem a referência "MC3090-Z".

Fig. 7 - Leitor e PDA Motorola MC3090-Z.

De entre as várias características deste produto, serão apresentadas as mais relevantes para este projecto. O seu preço encontra-se entre 1.700€ e 1.900€.

A sua antena utiliza uma nova tecnologia que alia uma polarização linear a uma polarização circular, obtendo deste modo as vantagens das duas opções. No capítulo 4.4, estes conceitos são explicitados com maior clareza.

Tem uma distância de leitura de aproximadamente 3 metros. Corre um sistema operativo *Windows Mobile 6.1*, com 128 Megabyte de memória RAM (*Read Access Memory*) e 1 Gigabyte de memória

Flash (memória onde se guardam as aplicações). Possui um botão físico para activar a leitura da antena e pesa 650 gramas. Possui certificados que atestam a sua resistência a choques, quedas, poeiras, salpicos etc., facto importante para o seu uso neste projecto. Inclui um teclado completo que poderá ser configurado.

Funciona na gama de frequências 902 a 928 MHz portanto as *tag*s a usar serão desta frequência (UHF).

Além disto, tem a capacidade de estar conectado á base de dados por vários meios, tais como 3G e *Wi-Fi*, actualizando assim em tempo real a base de dados que estará disponível para o gestor de manutenção.

Em anexo, encontra-se a ficha técnica deste produto.

Fig. 8 - Leitor e PDA da System Limited ref. CS 101.

Outro exemplo é o produto da Convergence Systems Limited, CS 101. Este difere do leitor da *Motorola* em alguns aspectos, mas o mais relevante seria o seu peso de 1 quilograma, o sistema operativo Windows CE, e a sua distância de leitura que é de 7 a 11 metros. O seu preço está entre 1.360€ e 1.750€.

Realça-se que o limite de distância de leitura está associado a uma limitação normativa, que limita a intensidade de ondas consoante a sua frequência.

A escolha do *tag* é ainda mais ampla. Existe uma enorme variedade de *tags* que podem ser usados neste sistema e a sua escolha dependerá do elemento a identificar. Isto deve-se a factores tais como, por exemplo, um *tag* num elemento móvel precisar de ser resistente a impactos enquanto que um *tag* num elemento fixo não necessita desta propriedade.

As propriedades do material onde estará o *tag* são também relevantes para a sua escolha, não esquecendo o facto que o mesmo leitor tem a capacidade de ler uma grande variedade de *tags*, desde que funcionem na mesma frequência.

Ainda assim mostram-se as características de um *tag* que poderá ser utilizado. Este *tag* é da empresa *Avery Denninson* e tem a referência AD-223. A sua referência vem do formato da antena, que pode ser aplicado a vários formatos de *tag*.

Fig. 9 - Formato da antena do *tag* da Avery Denninson ref. AD-223.

Tem as dimensões 95x8.15 mm, funciona entre as temperaturas -40ºC a 85ºC e funciona entre as frequências 860 a 960 MHz.

5.7. IMPLEMENTAÇÃO

Um dos factores que pode influenciar bastante no desenvolvimento da aplicação desta tecnologia na manutenção de edifícios é a sua adopção em massa pela indústria relacionada com a construção civil. Como referido em 5.4, a adopção deste sistema num edifício em construção é sem dúvida mais simples e económico do que num edifício em fase de utilização. Isto deve-se ao facto de que na aplicação de *tags* durante a fase de construção a elementos fabricados na obra não se antevêem grandes dificuldades.

A identificação de componentes estruturais poderá ser efectuada se necessário. Para isso ter-se-ia que colocar um *tag* na fase de betonagem, este que obrigatoriamente será impermeável e resistente a altas temperaturas. Em elementos não estruturais os tags têm várias opções de colocação, dando como exemplo uma parede de alvenaria onde o *tag* poderia ficar entre tijolos.

Os elementos pré-fabricados poderiam ser identificados já no processo de fabricação. A adopção desta medida teria de ter em conta o tipo de produto pois, no caso de um ar condicionado poder-se-ia facilmente integrar um *tag* no seu interior mesmo depois de construído mas, no caso de uma porta, este deveria ser integrado no processo de fabrico. Dadas as características de produção em série destas indústrias e o custo dos *tags*, a sua aplicação não aumentaria significativamente o custo final do elemento, tornando assim a identificação dos elementos mais simples e económica.

5.8. CONCLUSÕES

Depois da apresentação de um cenário ideal no ponto de vista do autor, da devida correlação com a tecnologia, consideração económica e das variantes de localização antevê-se um sistema de identificação RFID com as seguintes características: todos os elementos fontes de manutenção têm integrado um *tag* RFID, que pode ser lido a distâncias de 3 a 10 metros. As informações básicas e registos sobre o elemento podem ser obtidos no momento e com rapidez.

6
SOFTWARE

Neste capítulo antevê-se um antevê-se as propriedades e características de um software que o autor considera ideal.

6.1. CONSIDERAÇÕES INICIAIS

Como referido, por uma questão de longevidade dos *tag*s e menor risco de perda de informação, concluiu-se que seria necessária a criação de uma base de dados para trabalhar bilateralmente com a memória interna do *tag* ou mesmo apenas com a informação de identificação do *tag* (número único).

Esta opção além de permitir que o gestor de manutenção tenha toda a informação centralizada, podendo verificar a informação associada a um elemento sem necessidade de ir até ele, permite também que o operário obtenha a mesma informação na proximidade do elemento.

A estrutura da base de dados é dada por entidades, atributos e associações entre eles. As entidades são um agrupamento de atributos que só a elas diz respeito. São então criadas tabelas cujo nome são as entidades e as suas colunas são os seus atributos. As associações são, como o nome permite concluir, interacções entre essas tabelas.

É nas associações que reside a importância das bases de dados, pois podemos cruzar dados de várias tabelas, algo que, sem a ajuda informática, seria um processo demorado.

Para termos acesso à informação contida nessas tabelas, basta fazer uma pergunta ao servidor, pedindo as informações que queremos visualizar, podendo ir de uma simples pergunta, que seria de fácil resposta mesmo sem bases de dados, até às perguntas que envolvem vastas informações cruzadas.

Como estas perguntas são feitas numa linguagem informática, não são acessíveis ao público-alvo deste projecto. Logo existe a necessidade da criação de um software que permita a interacção do utilizador com a base de dados, obtendo de forma rápida e inequívoca as informações que necessita.

6.2. VISÃO DE SOFTWARE IDEAL

Tendo em consideração a evolução das tecnologias de sistemas de informação nas últimas décadas e as linguagens de programação avançadas actualmente disponíveis, não se prevêem dificuldades relevantes no que diz respeito à concepção do software necessário para dar resposta às exigências do projecto em questão.

Neste capítulo, faz-se referência a algumas das potencialidades que um software, aliado à tecnologia de identificação RFID, poderiam trazer.

6.2.1. VISÃO DO OPERADOR

Em primeiro lugar, quando a antena lê os *tags* à sua volta, criaria automaticamente uma lista de elementos que estão ao alcance da antena.

Escolhido um dos elementos ter-se-ia acesso a informações básicas sobre ele. Essas informações básicas seriam algo como nome do elemento, categoria, localização, estado do elemento, custo, garantia, informações sobre o projecto do elemento, empresa fabricante, empresa de montagem, etc.

A categoria é um agrupamento de elementos cujos processos de manutenções são iguais.

A localização identifica onde se encontra o elemento. Muitos edifícios possuem já um esquema de código de localização, que é habitualmente numérico, indicando o andar e o número da sala, podendo assumir varias formas.

O estado do elemento identifica se está a funcionar correctamente ou se está com alguma acção pendente como uma manutenção periódica por fazer ou uma solicitação de manutenção sobre o elemento.

As informações sobre o projecto poderiam incluir uma vasta gama de informações, tal como um ficheiro de desenho *AutoCad* para o caso de um corte de uma parede. Pode também ser uma informação alterável. No caso de um quadro eléctrico, por exemplo, sempre que existisse uma alteração ao projecto inicial seria actualizada. Esta informação é bastante útil no caso de um grande edifício em que há vários operadores de manutenção. Desta forma, não haveria tempos perdidos a tentar perceber o estado actual do quadro eléctrico. No caso de componentes electromecânicos, tal como elevadores ou ar condicionados, poderia ter um manual de utilização. Qualquer informação de projecto que fosse relevante para o operador seria colocada neste campo.

Na informação das empresas, existiriam informações básicas como nome da empresa, página Web, correio electrónico, morada, contactos, etc.

Depois de identificado o elemento ter-se-ia acesso a todas as operações de manutenção cuja execução está prevista (informações como manuais de limpeza, inspecção, pró-acção e substituição). Nestas informações estariam descritas a tarefa, o procedimento completo, o responsável, a periodicidade e ainda o seu custo previsto associado. No caso de existir uma patologia no elemento, existiria também uma lista das correcções mais comuns ao elemento, o seu procedimento adequado e também o custo previsto associado.

No caso de o operador querer executar alguma das tarefas, bastaria escolher uma e pressionar um botão que o conduziria à abertura de um novo registo nesse elemento. Esse registo iria ficar gravado numa lista de registos associada a esse elemento e poder-se-ia consultar esse registo sempre que necessário.

Cada registo grava uma série de informações a ele relacionado, tal como a data em que foi executado, qual a tarefa, o custo, o estado da tarefa, o tipo de tarefa, o tempo que demorou a ser executada, a identificação do operador que a executou e um campo livre de observações.

No caso de ser necessária a substituição de algum componente para repor a funcionalidade ao elemento, o operador poderia requisitar este directamente pelo software. Poderia funcionar directamente com a empresa fornecedora, enviando automaticamente um pedido de compra com a referência da peça ou poderia enviar um e-mail ao gestor da manutenção do edifício para que este resolvesse da maneira correcta.

O software deveria de ter um interface simples e intuitivo, de modo que operadores de empresas exteriores, que realizem operações de manutenção no edifício, pudessem utilizar o equipamento apenas com uma curta formação.

6.2.2. VISÃO DO GESTOR DE MANUTENÇÃO

Neste momento, a base de dados, além de registar todas as acções executadas nos elementos, também tem a periodicidade das acções supostas de executar nesse elemento.

Através de um cruzamento de dados, o software seria capaz de ter uma previsão de todos os trabalhos a serem realizados num determinado período de tempo, isto é, tanto se poderia ter um plano de trabalho geral previsto para o próximo ano como um plano de trabalho mais detalhado para uma semana. Este plano de trabalhos poderia ser desenvolvido tendo em conta as diferentes perspectivas de decisão.

Poder-se-ia decidir a execução de todos os trabalhos pendentes no mesmo local ou então efectuar a mesma tarefa a todos os elementos presentes no edifício.

Poder-se-ia também relacionar às várias tarefas a executar o operador por elas responsável, enviando essa informação ao operador, ficando ele assim a saber o seu plano de trabalhos.

Associado a isto, o software conseguiria também elaborar um orçamento previsto para esse intervalo de tempo, no caso de só se cumprirem as operações de manutenção periódicas.

O software teria também a capacidade de prever, por meios estatísticos, avarias em elementos. Esta capacidade seria possível comparando as correcções efectuadas a vários elementos iguais à sua idade, nível de utilização ou outros factores importantes, podendo assim estimar um período de tempo para o qual poderá requerer uma manutenção específica.

Com base nestas duas informações, em que se concilia os custos das operações efectivamente necessárias e das acções previstas estatisticamente, o software poderia estimar um orçamento que estaria mais próximo do valor real.

Com a obrigação de o operador ir até ao elemento para poder abrir um registo de manutenção (para o *tag* ficar ao alcance da antena RFID), o gestor de manutenção teria de facto a certeza que o operador pelo menos esteve próximo do elemento.

Supondo dois elementos com a mesma função e o mesmo nível de utilização, mas de empresas fabricantes diferentes, o gestor de manutenção teria acesso a informações que permitiriam concluir qual dos elementos implicaria uma maior despesa na sua fase de utilização.

Toda esta informação, disposta de uma maneira normalizada e globalizada, permitiria a criação de uma lista de desempenho de elementos de vários fabricantes. Esta informação poderia ajudar um cliente a escolher um produto com base na sua performance em fase de utilização e, claro, todos os outros factores importantes ao produto (tal como preço, funções, etc.).

Esta informação não seria apenas proveitosa para o cliente que vai comprar um produto novo mas também para o gestor de manutenção que irá substituir um produto.

Da mesma maneira, as empresas fabricantes poderiam aproveitar o acesso a esses dados, quer para melhorias do seu produto, quer também para referenciar uma potencial qualidade.

6.2.3. VISÃO DO UTILIZADOR DO EDIFÍCIO

Os utilizadores do edifício são de facto os que necessitam de uma maior satisfação. Uma das medidas a implementar seria um suporte electrónico via internet para que o utilizador pudesse enviar uma reclamação sobre algum elemento do edifício.

Essas reclamações poderiam ser, por exemplo, o mau funcionamento de uma porta ou uma lâmpada fundida. O suporte electrónico criaria uma ficha de reclamação a ler directamente pelo gestor de manutenção, que encaminharia para o operador ou, numa perspectiva diferente, dependendo do tipo de queixa, iria directo para o operador responsável.

Assim que fosse aberto um registo para essa reclamação e fosse dado como executado, o utilizador que enviou a reclamação receberia um e-mail a actualizá-lo da sua reclamação. Isto aumentaria a confiança que o utilizador deposita na equipa de manutenção, diminuindo os casos de patologias que realmente afectem o utilizador.

6.3. CONCLUSÕES

Dadas as perspectivas de um software ideal para a manutenção de edifícios, prevê-se que este seja de grande utilidade. Várias características que o software teria a capacidade de fazer automaticamente seriam extremamente morosas se efectuadas manualmente. O software para uma boa aplicação teria de ter um funcionamento e uma interface gráfica bastante intuitivos. Algumas funções só teriam um bom desempenho com a globalização deste sistema.

A opinião do autor é que, para um funcionamento integral do software, seria necessária a normalização pela Organização Internacional de Normalização, que visasse um formato específico de informações. Essa normalização passaria por especificar as informações necessárias de cada produto fabricado. Estas informações estariam presentes na memória interna do *tag* do elemento e o software adicionaria o elemento à base de dados pela sua leitura directa. Ao adicionar o elemento, acrescentava também informações sobre os manuais de manutenção que constariam na memória do *tag*. A partir desse momento, trabalhar-se-ia apenas com a base de dados do próprio edifício, não adicionando mais informação ao *tag*. As informações de cada edifício seriam posteriormente actualizadas automaticamente para um site internacional onde constariam edifícios, assim como empresas de elementos de todo o mundo. Nesse site ter-se-ia acesso a estimativas e médias de performance de cada produto, assim como preços de compra e de manutenção. Sem esta globalização, estas funções não seriam possíveis pelo facto de que seria necessário um número significativo de amostras para as previsões estatísticas se aproximarem da realidade.

7
CONCEPÇÃO DO SISTEMA

7.1. INTRODUÇÃO

A concepção do sistema informático que suporta este projecto não visa a sua aplicação imediata tratando-se apenas de uma prova de conceito[3]. Depois de detalhadas as visões óptimas do sistema RFID e do software a ele associado, ir-se-á criar um sistema de teste. Para isso houve a necessidade de adquirir um leitor RFID, criar uma base de dados e uma aplicação.

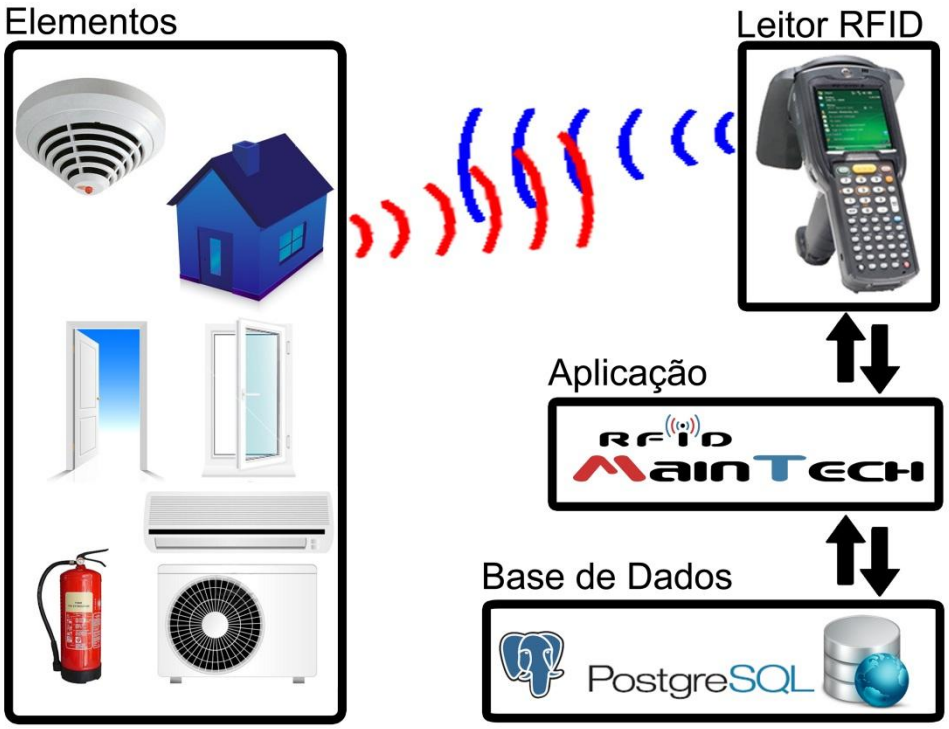


Fig. 10 - Esquema de funcionamento do sistema.

[3] Prova de conceito ou *Proof of* Concept em Inglês, é uma implementação, em geral resumida ou incompleta, de um método ou de uma ideia, realizada com o propósito de verificar que o conceito ou teoria em questão é susceptível de ser explorado de uma maneira útil. A prova de conceito é considerada habitualmente um passo importante no processo de criação de um protótipo realmente operativo.

7.2. SISTEMA RFID

O leitor usado para este projecto foi cedido no período de desenvolvimento deste trabalho pela empresa Telemax - Telecomunicações e Electrónica, Lda. É um leitor direccionado para *tags* de cartões de plástico. A sua frequência de operação é 12,5kHz o que embora não corresponda à frequência escolhida no capítulo 5.6, é suficiente para a prova de conceito que se pretende desenvolver.

Com este leitor é possível ler *tags* através de alguns materiais, mas a sua distância de leitura máxima é extremamente reduzida, menos de dez centímetros, devido à sua frequência de operação.

Os *tags* compatíveis com este leitor podem ter vários formatos, entre eles; cartões plásticos, forma de moeda autocolante, porta-chaves, etc. Apesar dos *tags* não poderem ser lidos a vários metros de distância conseguem provar o conceito.

Fig. 11 - NEXUS ref.BMU-01 Leitor usado neste projecto.

7.3. BASE DE DADOS

7.3.1. INTRODUÇÃO

Antes de tudo, é necessário esclarecer, de um modo sucinto, o que é uma base de dados.

Base de dados é um conjunto de registos informáticos dispostos numa estrutura regular que possibilita a reorganização dos mesmos e a produção de informação. Uma base de dados normalmente agrupa registos utilizáveis para um mesmo fim. [22]

Uma base de dados relacional tem 3 noções que a constituem: as Entidades, os Atributos e as Associações.

Atente-se num exemplo simples para mais sucintamente se clarificarem as noções. Considere-se uma lista de Empregados numa empresa, que tem os valores: Nome e Número BI. Considere-se ainda uma lista de Encargos nessa empresa com os valores: Nome, horas de trabalho por semana e salário. Cada trabalhador da empresa está ligado a um cargo.

Neste exemplo, Empregados e Encargos são as Entidades. Nome e Data de nascimento são Atributos da entidade Empregados. A relação existente entre as Entidades Trabalhador e Encargo constitui uma Associação.

Com isto, pode obter-se uma lista de todos os trabalhadores, assim como o dinheiro gasto por mês na empresa, fazendo assim cruzamento de dados entre duas entidades através da sua associação.

7.3.2. ENTIDADES

Neste projecto iremos ter as seguintes Entidades:

- Elemento
- Empresa
- Registo
- Categoria
- LIPS

A entidade **Elemento**, tal como o nome indica, é a lista de todos os elementos identificados no projecto independentemente da sua categoria ou localização. É uma lista misturada mas com as informações normalizadas.

A entidade **Empresa** é uma lista de informações de todas as empresas que estão ligadas ao projecto em questão.

Registo é uma lista em que estão presentes todas as acções de manutenção efectuadas aos elementos. Funciona como um cadastro que poderá ser utilizado como consulta pelo operador ou para o fim de cálculos automáticos pelo software.

Categoria é a lista de todas as categorias presentes no projecto, tal como porta, parede, cobertura, etc. Esta entidade foi criada para que se possam agrupar elementos cujos processos de manutenções sejam iguais. Também dá a possibilidade ao operador de executar tarefas no software direccionadas a apenas uma categoria de elementos.

LIPS é a sigla de Limpeza, Inspecção, Pró-Acção e Substituição. É uma lista de todas as operações de manutenção necessárias aos elementos de um dado projecto. Apesar de não estar associado á sigla LIPS, as Correcções também se encontram nesta lista. Estas informações estão presentes nos manuais de manutenção de cada elemento. Muitos elementos já possuem um manual de manutenção que vem da empresa fabricante, mas ainda assim existe um grande número de trabalhos realizados sobre os processos de manutenção dos mais variados elementos.

7.3.3. ASSOCIAÇÕES

Um Elemento irá ter uma Empresa fabricante e uma Empresa de montagem. Cada um pertence a uma Categoria e poderá ter uma infinidade de Registos. Uma Categoria irá ter uma série de acções de manutenção associadas a ela (LIPS).

Com isto, na Fig. 12 apresenta-se uma primeira visão da relação Entidade-Associação.

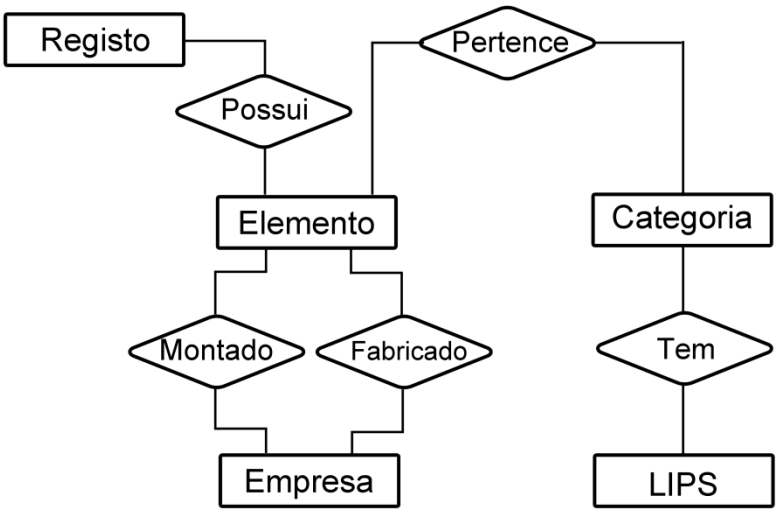


Fig. 12 - Entidades e Associações.

Como se depreende da Fig. 12, as entidades estão inseridas em rectângulos e as suas associações em losangos.

Temos portanto as seguintes associações: Montado, Fabricado, Possui, Pertence e Tem.

7.3.4. CARDINALIDADE

Analisa-se agora a cardinalidade de cada associação. A cardinalidade é uma característica que define se o tipo de relação entre as entidades é de um para muitos ou muitos para muitos. Dando um exemplo: a relação entre empregados e encargos em que um empregado só pode ter um encargo mas um encargo pode ter muitos empregados, logo a relação entre empregados para encargos é de muitos para um ou, na sua simbologia, de N:1.

Associação **Montado** e **Fabricado**: Um Elemento pode apenas ter uma Empresa fabricante. Em contrapartida, uma Empresa pode fabricar vários Elementos. Com isto temos uma cardinalidade entre Elemento e Empresa de N:1 (muitos Elementos para uma Empresa fabricante). Da mesma maneira se tem a cardinalidade de Empresa de montagem, associando assim um Elemento a duas Empresas diferentes, uma de montagem e uma de fabrico com a mesma cardinalidade.

Associação **Possui**: Um Elemento pode possuir uma infinidade de Registos, e cada um desses Registos pertence a apenas um Elemento sendo assim a cardinalidade entre Elemento e Registo de 1:N (um Elemento para muitos Registos).

Associação **Pertence**: Um Elemento pertence a apenas uma Categoria e uma Categoria pode ter uma infinidade de Elementos sendo portanto uma cardinalidade de N:1 (muitos Elementos para uma Categoria).

Associação **Tem**: A tabela LIPS, onde se encontram uma série de procedimentos de manutenção, irá ter uma cardinalidade com a tabela Categoria de N:N (muitos LIPS para muitas Categorias). Dando um exemplo para melhor se entender, imagine-se que existe a Categoria “Porta de madeira X de

fechadura manual" e a Categoria "Porta de madeira X de fechadura automática". As duas categorias têm processos de manutenção iguais, como por exemplo a substituição das anilhas da dobradiça. Mas os processos de manutenção da fechadura irão ser diferentes para cada categoria. Portanto existem processos de manutenção incluídos no elemento LIPS que serão partilhados por várias Categorias.

Dadas as cardinalidades, teremos agora o seguinte esquema Entidade-Associação da Fig. 13.

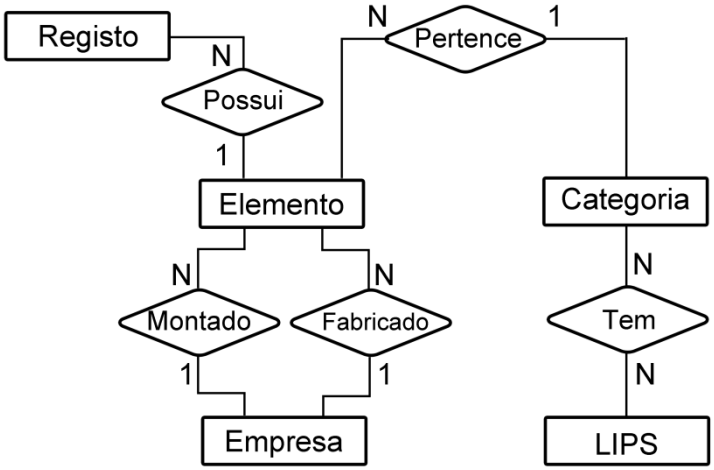


Fig. 13 - Cardinalidade entre Associações.

7.3.5. Atributos

Os atributos, como explicado em 7.3.1, são valores que definem propriedades de uma entidade. É de importante a inclusão de todos os atributos de modo a que posteriormente se possam realizar operações para as características projectadas do software.

Com isto indicam-se agora os atributos de cada entidade, o código correspondente na base de dados assim como uma explicação de cada atributo.

7.3.5.1. Elemento

Elemento: código RFID, Nome, Descrição, Data de Compra, Localização, Elemento de Projecto, Custo e Garantia. Os códigos na base de dados são respectivamente: *cod_rfid, nome, descricao, dataelemento, cod_local, proj, custo, garantia.*

O **Código RFID**, tem a função de identificar o elemento. Este terá o valor do número único de identificação contido na memória interna do *tag*. É definido como a Chave Primária[4] da categoria Elemento, podendo com este número localizar inequivocamente um elemento específico na base de dados.

[4] Chaves primárias (em inglês Primary Keys ou PK) sob o ponto de vista de uma base de dados relacional, referem-se aos conjuntos de um ou mais campos, cujos valores, considerando a combinação de valores de todos os campos da tupla, nunca se repetem e que podem ser usadas como um índice para os demais campos da tabela do banco de dados. Em chaves primárias, não pode haver valores nulos nem repetição de tuplas. [21]

O **Nome** tem como objectivo ser uma informação que possibilite o operador identificar o elemento e que, ao contrário do código RFID, seja legível e compreensível ao operador. Permite, por exemplo, especificar se o elemento lido é a fachada Sul ou a fachada Este.

A **Data de Compra** é um atributo que define a data de entrada em serviço do elemento. Esta data serve não só para consulta do operador, mas também para cálculos automáticos associados à garantia assim como serve de base para comparação com as periodicidades dos processos de manutenção.

O atributo **Localização**, é fundamental para que se se identifique qual o edifício e a sua localização dentro deste. Esta informação será igual para todos os elementos dentro da mesma sala, podendo deste modo interagir com todos os elementos de um mesmo local apenas identificando um por RFID. Esta localização irá ser num formato de um código que consiste em 5 caracteres: O primeiro será uma letra identificando o edifício no caso de o projecto consistir em vários edifícios. O segundo será um sinal "+" ou "-", indicando respectivamente se o piso está acima ou abaixo do nível térreo do edifício. O terceiro carácter será um algarismo indicando numericamente o piso correspondente. Finalmente os dois últimos serão algarismos indicando da área, esta que poderá ser uma sala, um corredor, etc. No caso de um edifício possuir mais de 99 áreas numeradas poder-se-á usar uma numeração de área de 3 algarismos, ficando assim o código de localização com 6 caracteres.

Exemplifica-se o caso de duas identificações de localização:

- "Z+224"- Edifício Z, piso 2 acima do nível térreo e área com o número 24.
- "X-102" – Edifício X, piso 1 abaixo do nível térreo e área com o número 02.

O **Elemento de Projecto** tem como principal função disponibilizar um caminho para um ficheiro com as plantas e cortes do elemento em questão, ficheiro esse que pode ser do formato que o gestor de manutenção assim pretender. O formato PDF seria ideal para a utilização em computadores com reduzido poder de processamento assim como PDA's, mas poder-se-ia por exemplo também introduzir um ficheiro AutoCAD. Com esta informação o operador ficará a saber, por exemplo, qual o corte de uma parede ou de uma cobertura, assim como o trajecto de possíveis tubagens de electricidade, água ou de AVAC (Aquecimento, Ventilação e Ar Condicionado) que atravessem o elemento em questão.

O **Custo** será um atributo que especifica qual o preço de um determinado elemento. Dado que o custo de certos elementos, como uma parede ou uma cobertura, não necessita ser especificado, este campo pode não estar obrigatoriamente preenchido.

A **Garantia** será um valor numérico em anos. Associando esta à data de compra, poder-se-á calcular qual a data de término da garantia de cada elemento.

7.3.5.2. Empresa

Empresa: Nome, Número de Identificação Fiscal, Endereço Electrónico, Correio Electrónico, Morada, Telefone, Faxe, Nome do Contacto, Telemóvel do Contacto. *(nome, nif, site, email, morada, telefone, fax, nome_contacto e tlm_contacto)*

Os atributos da entidade **Empresa** têm o objectivo de dar acesso ao operador de dados relevantes sobre a empresa fabricante ou de montagem. Os atributos **NIF** e **Nome** são de preenchimento obrigatório. O primeiro é a Chave Primária da entidade Empresa, pois o NIF funciona como um número de bilhete de entidade da empresa. Os demais atributos desta entidade serão apenas de informação ao operador, conferindo-lhe o conhecimento de várias formas de contacto da empresa.

7.3.5.3. Registo

Registo: Custo, Estado da intervenção, tipo, observações, relatório e a data. *(custo, estado, tipo, observ, relatorio, data_reg)*

O atributo **Custo** tem como objectivo atribuir um custo a cada acção de manutenção executada. Este custo serve tanto para consulta do operador ou do gestor, como também para processos automáticos efectuados pelo programa.

O atributo **Estado** pode tomar 3 valores: **Programado/A Fazer**, **Em Execução** e **Executado**. Este atributo, tal como os 3 nomes indicam, define o estado actual do registo.

Devido ao facto de existirem acções de manutenção que têm um prazo de execução bastante longo, sentiu-se a necessidade de criar o Estado "**Em Execução**". O estado "**Programado/A Fazer**" foi incluído para que o gestor possa indicar, de um modo manual, um procedimento da manutenção que seja necessário fazer a um elemento. Isto pode acontecer por exemplo devido a uma reclamação. O Estado "**Executado**" é necessário não só para o gestor ter conhecimento de que uma tarefa foi executada, mas também para que o software possa efectuar um processamento de dados que necessite da confirmação de uma dada acção. A cada um deles foi atribuído o valor numérico 1, 2 e 3 respectivamente. Esta opção foi tomada devido à opção numérica ocupar de menos espaço na base de dados, não porque isso constitua um problema incontornável numa base de dados, mas sim por uma questão de performance das operações do software que envolvam o uso deste atributo.

O atributo **Tipo** é utilizado para atribuir uma classificação ao tipo da manutenção efectuada. Este atributo pode tomar um dos seguintes valores: **Limpeza**, **Inspecção**, **Pró-Acção**, **Substituição** e **Correcção**. Esta informação será utilizada não só para uma melhor visualização e compreensão dos registos associados a um elemento, assim como para ser utilizada automaticamente pelo programa para a execução de um Relatório de Actividades. De referir que, como o atributo **Estado**, estes valores são transformados em valores numéricos, respectivamente 1, 2, 3, 4 e 5, para uma melhor performance global.

O atributo **Observações** serve de campo onde o operador de manutenção possa escrever qualquer informação relevante sobre a acção efectuada. Especificam-se dois exemplos para uma melhor compreensão do atributo e em que casos se prevê a sua utilização: no caso de o operador efectuar alguma acção não listada nos processos de manutenção da base de dados, descrevendo o procedimento que seguiu; no caso de o operador não ter executado pelo procedimento especificado, esclarecendo o procedimento utilizado e a razão de tal.

O atributo **Relatório** é utilizado para, de um modo resumido, explicar a acção. Quando se executa uma acção de manutenção presente na entidade **LIPS**, o valor do atributo **Tarefa** (da entidade LIPS) tornar-se-á no valor do **Relatório** (da entidade **Registo**). Este atributo é da utilidade do gestor, pois especifica a tarefa executada, mas tem como principal objecto o seu uso numa das funcionalidades do programa executado.

O atributo **Data de Registo** é um valor que indica a data em que o registo foi criado. Esta é uma informação que além de permitir ao gestor ter conhecimento da data em que foi executada uma acção, é também utilizada em cálculos automáticos do programa.

7.3.5.4. Categoria

Categoria: Designação e Imagem. *(designacao, imagem)*

O atributo **Designação**, chave primária da entidade, tem como objectivo atribuir um nome a um grupo de elementos, em que pode tomar o valor de, por exemplo, "Extintor". Já o atributo **Imagem** tem como valor um caminho específico em formato de directório para que o programa possa carregar uma imagem relacionada com o elemento. O motivo do programa carregar a imagem correspondente à categoria é mais para que haja mais um meio para que o operador compreenda que está a lidar com o elemento pretendido, evitando assim alguma confusão que possa existir no elemento com que se está a comunicar. Existiria também a hipótese de associar o atributo **Imagem** ao Elemento mas esta opção foi descartada devido à obrigação de catalogar fotograficamente todos os Elementos. Deste modo, com a **Imagem** associada à entidade **Categoria**, apenas é necessária uma imagem de qualquer elemento pertencente à **Categoria**. Dá-se esta possibilidade porque dentro da mesma categoria, em geral, todos os elementos serão iguais ou bastante idênticos.

7.3.5.5. LIPS

LIPS: Tarefa, Responsável, Procedimento, Periodicidade e Custo. *(tarefa, responsavel, procedimento, periodicidade, custo)*

O atributo **Tarefa** tem como objectivo especificar de um modo sucinto a tarefa de manutenção descrita. Se o operador executar uma tarefa de manutenção presente na entidade **LIPS**, o valor deste atributo será o que o atributo **Relatório** da entidade **Registo** tomará.

O atributo **Responsável** tem como função identificar o operador ou empresa responsável pelo processo de manutenção associado.

O atributo **Procedimento** é necessário para descrever o modo como o operador, ou o responsável, deve actuar na execução deste processo. Deste modo o operador ao executar alguma acção de manutenção a um elemento, tem todos os dados necessários para proceder da maneira correcta.

O atributo **Periodicidade** tem um valor numérico em anos. Este valor define a periodicidade que um processo de manutenção tem e é utilizado pelo programa para a funcionalidade **Verificação de Manutenções**.

O atributo **Custo** é também um valor numérico mas em Euros. Este valor define um custo estimado do processo de manutenção correspondente. O programa usa-o, por exemplo, para estimar custos de manutenção num dado período de tempo.

7.3.6. Modelo Entidade-Associação

Estando definidas todas as propriedades da base de dados, é agora possível esquematizar toda a informação num modelo entidade-associação.

Os atributos são inseridos em elipses como se pode verificar pela Fig. 14.

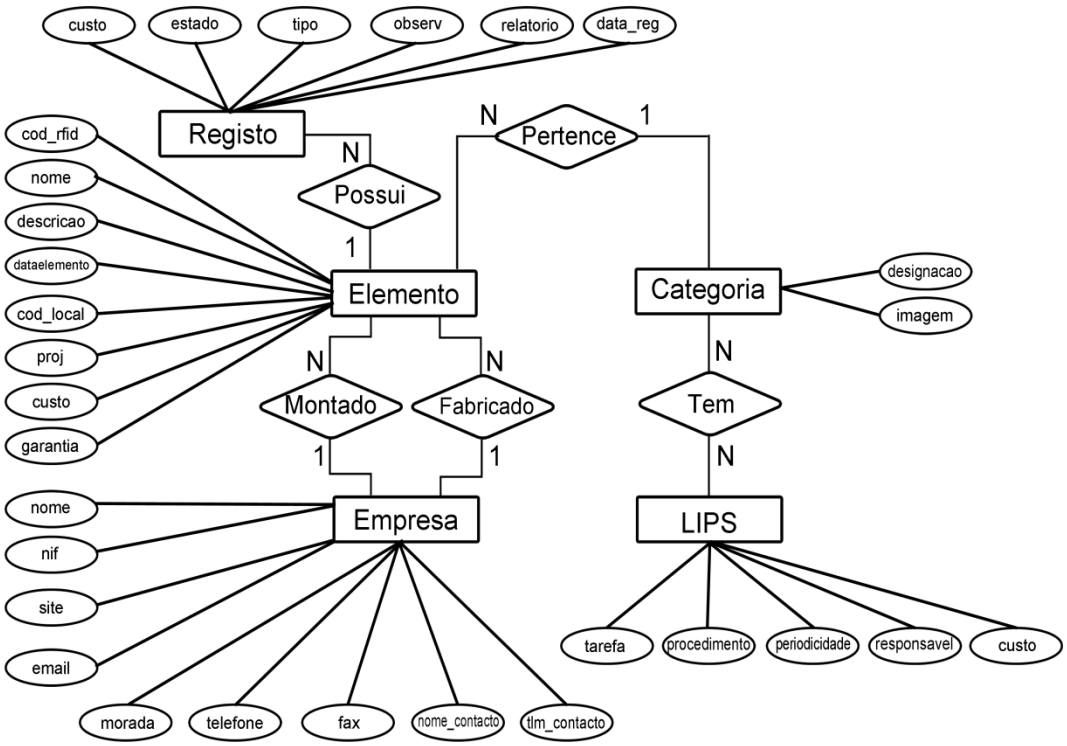


Fig. 14 - Modelo Entidade-Associação completo.

7.3.7. Sistema de Gestão de Base de Dados

Uma base de dados é geralmente acedida e mantida por meio de um software conhecido como Sistema Gestor de Base de Dados (SGBD)[5]. O modelo de bases de dados mais adoptado hoje em dia é o modelo relacional, onde as estruturas têm a forma de tabelas, compostas por registos (linhas) e colunas.

Cada linha de uma base de dados, formada por um conjunto de colunas, representa um registo. Os registos não precisam necessariamente de conter dados em todas as colunas, os seus valores podem ser nulos.

Existe actualmente uma grande quantidade de softwares capazes de satisfazer as necessid\ades deste projecto. Exemplificando, os mais conceituados são: Microsoft Visual FoxPro, Microsoft Access, dBASE, FileMaker, MySQL,Intpró, PostgreSQL, Firebird, Microsoft SQL Server, Oracle, Informix, DB2, Caché, Sybase, etc. [22]

Neste projecto decidiu-se utilizar o software PostgreSQL, pelo facto de ser um software com uma licença BSD (Berkeley Software Distribution), que significa que o código fonte é aberto à comunidade e não tem custos associados. Além disso é um software que pode ser executado em vários sistemas operativos.

[5] Um Sistema de Gestão de Bases de Dados, (SGBD) é um conjunto de programas que permitem armazenar, modificar e extrair informação de uma base de dados. Há muitos tipos diferentes de SGBD. Desde pequenos sistemas que funcionam em computadores pessoais a sistemas enormes que estão associados a *mainframes*. Um Sistema de Gestão de Base de Dados implica a criação e manutenção de bases de dados, elimina a necessidade de especificação de definição de dados, age como interface entre os programas de aplicação e os ficheiros de dados físicos e separa as visões lógica e de concepção dos dados.

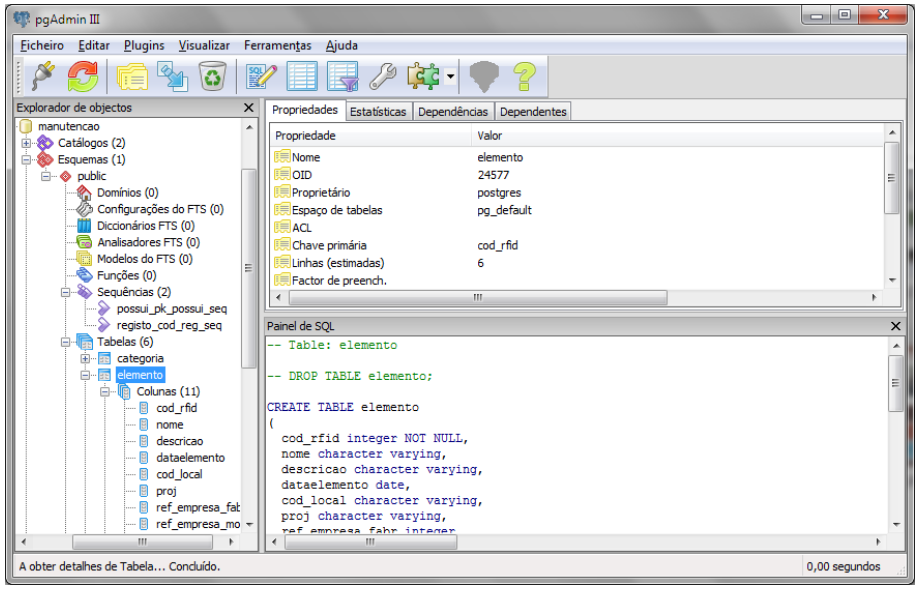


Fig. 15 - Ecrã de administrador do software PostgreSQL.

Para a criação no software PostgreSQL da base de dados demonstrada nos capítulos 7.3.1. a 7.3.6., foi necessário realizar algumas operações adicionais.

Foi criada uma nova Base de Dados com o nome de "**Manutencao**". Dentro dessa base de dados foram criadas as entidades **Categoria**, **Empresa**, **Elemento**, **LIPS** e **Registo**.

Todas as entidades devem ter uma Chave Primária para permitir a sua rápida localização assim como conferir a propriedade de se poder associar várias tabelas.

As entidades **Elemento**, **Empresa** e **Categoria** têm as suas Chaves Primárias definidas no capítulo 7.3.5. Já nos casos das entidades **Registo** e **LIPS** teve-se a necessidade de proceder à criação de novos atributos identificadores. Na entidade **Registo**, inseriu-se o atributo *cod_reg*, um atributo numérico simples e crescente. No caso da entidade **LIPS** foi necessário criar um sistema mais complexo. Foi criado o atributo *cod_lips* em que o seu valor é diferente para cada tipo de manutenção diferente. Exemplificando, o valor do atributo *cod_lips* nas manutenções de limpeza serão do género l1, l2, l3, etc. Logo têm-se diferentes tipos de contagem para a chave primária de **LIPS** para cada um dos tipos de manutenção, limpeza (l1,l2,l3,…), inspecção (i1,i2,i3,…), pró-acção (p1,p2,p3,…), substituição (s1,s2,s3,…) e correcções (c1,c2,c3,…).

Cada associação, dependendo do seu tipo de cardinalidade, terá uma que ter um sistema que permita a ligação entre as tabelas. Para isto, além dos atributos referidos no capítulo 7.3.5., foi necessário criar atributos referentes às associações definidas em 7.3.3. Estes atributos são designados de "Chave Externa" (*Foreign Key*).

Nas associações **Montado** e **Fabricado** dada a cardinalidade ser, "de uma Empresa para vários Elementos", criou-se um atributo na entidade **Elemento** para referir cada associação. Isto é, foram criados dois atributos, *ref_empresa_mont* e *ref_empresa_fabr*, que irão ter os valores do NIF (da entidade Empresa) associado à Empresa de Montagem e à Empresa Fabricante respectivamente.

Na associação **Pertence,** a cardinalidade mantém-se a mesma, "de uma categoria para vários elementos", logo tornando-se necessário criar um atributo na entidade Elemento, *ref_ele_cat*, cujo valor é a do atributo Designação da entidade Categoria.

A associação **Possui**, que tem uma cardinalidade, "de vários registos para o mesmo elemento", implica a criação de um atributo na entidade Registo, *ref_ele*, que terá o valor do atributo Código RFID da entidade Elemento.

Já na associação **Tem**, dada a cardinalidade ser, "de muitos processos de manutenção para muitas categorias", foi necessário criar uma nova tabela, em que se referenciaram as duas entidades. Esta nova tabela tem o nome de **Possui** (não confundir com a associação Possui) e os seus atributos são os seguintes: *ref_cat*, *ref_lips*, *pk_possui*. O primeiro tem o atributo Designação da entidade Categoria, o segundo tem o atributo do atributo *cod_lips* da entidade LIPS, e o terceiro é a chave primária de cada ligação entre Categoria e LIPS.

Para adicionar os atributos correspondentes às entidades criadas, o software tem a necessidade de definir qual o tipo de dados que cada atributo irá guardar.

Neste trabalho utilizaram-se os seguintes tipos de dados como atributos:

- Character Varying
- Integer
- Date
- Money
- Serial

Explica-se de seguida o significado de cada um dos tipos de dados assim como se exemplifica o seu uso.

Character Varying: Aceita qualquer letra ou algarismo podendo conter sinais de pontuação, acentuação, assim como espaços vazios. Nesta base de dados foi definido não utilizar nenhum limite da quantidade de texto em nenhum dos atributos com este tipo de dados. O nome de um elemento, ou a morada de uma empresa são exemplos de atributos associados a este tipo de dados.

Integer: Apenas aceita algarismos e não pode conter espaços entre eles. O Código RFID e o NIF de uma empresa são exemplos em que este tipo de dados é utilizado.

Date: Este tipo de dados apenas aceita um formato de data. Geralmente utiliza-se o modelo AAAA-MM-DD (Ano – Mês – Dia, por ex: 2007-05-03). A data de compra de um elemento ou a data de um registo de manutenção são dois exemplos deste tipo de dados.

Money: Este tipo de dados é específico para trabalhar com unidades monetárias. O custo do elemento e o custo estimado de uma acção de manutenção são os únicos atributos que lhe estão atribuídos.

Serial: Tem as mesmas propriedades que o tipo Integer. A sua única diferença é que este tipo é utilizado para ser uma *primary key* que não necessita de ser especificado pelo software. Este tem a propriedade de gerar um número automático baseado no último número de entrada da lista.

Sintetizam-se alguma das informações deste capítulo na Tabela 1, onde se descrevem as entidades, os elementos associados a elas e o tipo de dados de cada elemento. Referenciam-se os atributos que têm as propriedades de Chave Primária (PK) e Chave Externa (FK). Para os que possuem a propriedade FK descrimina-se ainda o seu destino de ligação de um modo "Entidade(atributo)".

Tabela 1 - Tabela de Entidades, Atributos, Tipos de dados e Constrições.

Entidade	Atributo	Tipo de dados	Constriçoes
Categoria	designacao	character varying NOT NULL	PK
	imagem	character varying	
elemento	cod_rfid	integer NOT NULL	PK
	nome	character varying	
	descricao	character varying	
	cod_local	date	
	proj	character varying	
	custo	money	
	garantia	integer	
	ref_empresa_fabr	integer	FK empresa(nif)
	ref_empresa_mont	integer	FK empresa(nif)
	ref_ele_cat	character varying	FK Categoria(designacao)
empresa	nome	character varying	
	nif	integer NOT NULL	PK
	site	character varying	
	email	character varying	
	morada	character varying	
	telefone	integer	
	fax	integer	
	nome_contacto	character varying	
	tlm_contacto	integer	
lips	cod_lips	character varying NOT NULL	PK
	tarefa	character varying	
	responsavel	character varying	
	procedimento	character varying	
	periodicidade	integer	
	custo	money	
possui	ref_cat	character varying	EK Categoria(designacao)
	ref_lips	character varying	EK lips(cod_lips)
	pk_possui	serial NOT NULL	PK
registo	custo	integer	
	estado	integer	
	tipo	integer	
	observ	character varying	
	relatorio	character varying	
	data_reg	date	
	ref_ele	integer	FK elemento(cod_rfid)
	cod_reg	serial NOT NULL	PK

7.4. SOFTWARE

7.4.1. INTRODUÇÃO AO LAZARUS

O desenvolvimento do software para este projecto será feito em Lazarus v.0.90.30. Lazarus é um ambiente de desenvolvimento integrado (*Integrated Development Environment - IDE*) desenvolvido para o compilador Free Pascal. O software tem como objectivo ser compatível com o Delphi, outro software IDE e, ao mesmo tempo, suportar diversas arquitecturas e sistemas operacionais.

IDE, do inglês *Integrated Development Environment* ou Ambiente Integrado de Desenvolvimento, é um programa de computador que reúne características e ferramentas de apoio ao desenvolvimento de software com o objectivo de agilizar este processo. Geralmente os IDEs facilitam a técnica de RAD (de *Rapid Application Development* ou Desenvolvimento Rápido de Aplicativos), que visa a maior produtividade dos desenvolvedores.

Free Pascal é um compilador de Object Pascal que pode ser executado em Linux, Windows, OS/2, Mac OS tradicional, Mac OS X, ARM, BSD, BeOS, DOS e mais. Ele foi desenhado para compilar código com a sintaxe do Delphi ou dos dialectos Pascal do Macintosh e gerar executáveis para diferentes plataformas a partir de um mesmo código-fonte.

Um compilador é um programa de computador (ou um grupo de programas) que, a partir de um código fonte escrito em uma linguagem compilada, cria um programa semanticamente equivalente, porém escrito em outra linguagem, código objecto.

Tem-se a noção que o desenvolvimento de um software como o descrito no capítulo 6 estaria fora do âmbito deste projecto não só pela sua complexidade mas também pelo tempo que demoraria a ser executado. Com isto, limitam-se alguns dos parâmetros do software, e apenas se incluem os que na perspectiva do autor teriam uma grande relevância sem a necessidade de uma globalização do produto.

Este software foi optimizado para funcionamento num *laptop*, mas o seu código fonte pode ser compilado para funcionar em *tablets* e PDAs, sendo apenas necessário reorganizar o *layout* para melhor se ajustar a um ecrã pequeno e a um ambiente de ecrã táctil. Deste modo, todo o código pode ser reutilizado.

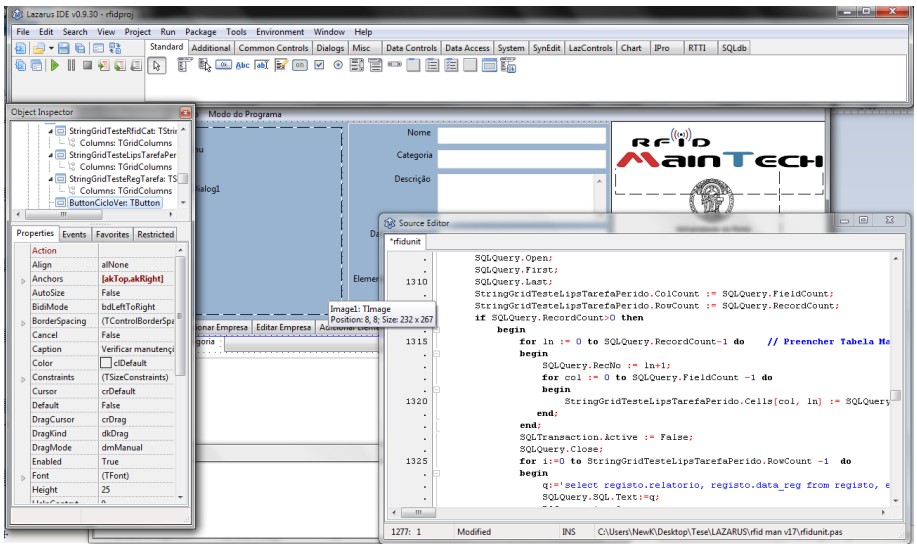


Fig. 16 - Ecrã de Lazarus em modo de programação.

O software Lazarus está divido em cinco ecrãs: **Lazarus IDE**, **Object Inspector**, **Source Editor**, **Messages** e **Form view**.

O ecrã **Lazarus IDE** é onde se encontra o menu superior, assim como todos os objectos que podem ser criados na aplicação actual. Estes objectos são os constituintes visuais do programa a ser executado. Entre eles constam botões, campos de edição de texto, componentes da ligação SQL, entre outros. Destes objectos, os utilizados neste projecto são: *TButton, TBitBtn, TLabel, TEdit, TMemo, TComboBox, TRadioGroup, TImage, TBevel, TStringGrid, TPageControl, TMainMenu, TOpenPictureDialog, TCalendar, TSQLQuery, TSQLTransaction e TPQConnection*.

TButton: Botão em formato de caixa normal do sistema operativo.

TBitBtn: Botão com imagem associada para melhor compreensão da sua função.

TLabel: Texto fixo.

TEdit: Caixa de edição de texto de uma só linha.

TMemo: Caixa de texto com possibilidade de múltiplas linhas.

TComboBox: Formulário de selecção única em formato de lista vertical.

TRadioGroup: Formulário de selecção em formato de círculos.

TImage: Caixa de imagem.

TBevel: Elemento decorativo em formato de caixa.

TStringGrid: Grelha de textos.

TPageControl: Menu em formato de pastas.

TMainMenu: Menu superior.

TOpenPictureDialog: Diálogo em formato de caixa para carregamento de imagens.

TCalendar: Caixa de edição de data com calendário acoplado.

TSQLQuery: Componente de ligação à base de dados SQL. Este componente especifica qual o *query* que o componente *TSQLTransaction* irá utilizar quando inicializado.

TSQLTransaction: Componente de ligação à base de dados SQL. Este componente está ligado enquanto estão activas ligações à base de dados.

TPQConnection: Abre um protocolo de comunicação à base de dados PostgreSQL. Tanto o objecto *TSQLQuery* como o TSQLTransaction necessitam deste objecto para conexão à base de dados.

Nos pontos seguintes explicitam-se as situações em que são utilizados estes objectos.

O ecrã **Object Inspector**, inclui a maior parte das propriedades de cada objecto de um modo visual. Dentro desta janela encontram-se duas páginas importantes: *Properties* e *Events*. Quando um objecto é adicionado à aplicação, este carece de uma personalização para a função a desempenhar. Dando um exemplo simples, um botão, uma das características presentes nas *Properties* é o nome que aparece dentro do botão (*caption*). Nos eventos definem-se procedimentos para todos os tipos possíveis de eventos associados ao objecto, um caso muito utilizado nos Botões é o evento *OnClick*. Clicando no evento, cria-se um procedimento associado ao acontecimento. Este procedimento terá de ser especificado no ecrã Source Editor na linguagem Pascal. De salientar que todas as propriedades de um objecto podem ser alteradas via Source Editor durante a fase de execução do programa (*Run Time*). Esta é uma propriedade importante visto que a propriedade *Text* (texto) presentes nas caixas de texto

(*TMemo* ou *TStringGrid*) dependem do elemento lido, logo esta propriedade necessita de ser definida em tempo de execução da aplicação.

O ecrã **Form View**, é onde está presente o *layout* da aplicação. Os objectos são colocados neste ecrã do seguinte modo: Clica-se no objecto a adicionar presente no ecrã **Lazarus IDE**, e de seguida clica-se a localização pretendida no ecrã **Form View**. Neste ultimo ecrã, definem-se também os tamanhos desejados para os objectos criados. Quando um objecto está seleccionado neste ecrã, o ecrã **Object Inspector** altera-se para as propriedades do objecto seleccionado.

O ecrã **Messages**, relata as informações procedentes da compilação do código. Quando se dá o comando de compilar o código fonte, o compilador tem uma característica de, em caso de algum erro, explicitar em que linha se encontra o erro e muitas das vezes caracteriza-lo. Este é o tipo de informação que surge neste ecrã.

Finalmente o **Source Editor**, é onde todo o código é escrito. Esta linguagem tem as potencialidades de executar ciclos Se, Para, Enquanto, etc. Ciclos estes que estão presentes em quase todas as linguagens de programação existentes. Este é um ecrã onde apenas tem texto, e será este texto que o compilador usará para compilar a aplicação.

Convém ainda explicar brevemente qual o procedimento de uma pergunta à base de dados. A pesquisa de um elemento na base de dados é feita através de *querys*. Dando um exemplo de uma *query* em que se desejam seleccionar o nome e respectivo local de todos os elementos presentes na base de dados, a *query* seria: "*SELECT nome, cod_local FROM elemento*". No caso de se querer especificar apenas um elemento, por exemplo, seleccionar o nome e o local do elemento com o código rfid igual a 1, a *query* ficaria: "*SELECT nome, cod_local FROM elemento WHERE cod_rfid=1*".

7.5. MAINTECH – APRESENTAÇÃO

Fig. 17 - Logotipo do software RFID MainTech.

Ao software desenvolvido foi atribuído o nome de **RFID MainTech**. Este software está conectado à base de dados gerido pelo software *PostgreSQL*. Este programa é constituído por um ecrã principal que possui um barra de menus superior (*TMainMenu*). Este ecrã principal está dividido em duas partes: a superior com informação fixa sobre o elemento lido e a inferior com informação variável (*TPageControl*).

De referir que o código fonte do programa consta com mais de 2300 linhas, inviabilizando a sua disponibilidade em papel. Desta forma, apresentam-se apenas algumas partes importantes do código para a compreensão do funcionamento do programa assim como algumas das suas interacções com a base de dados.

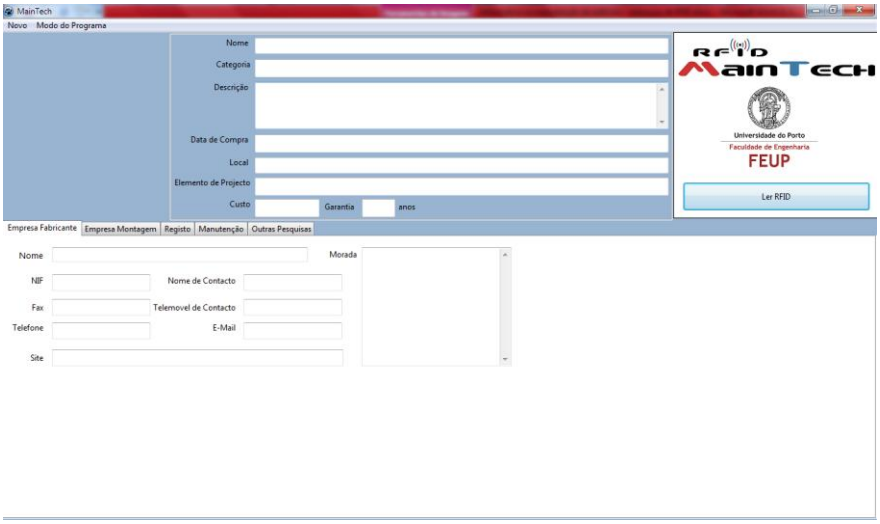


Fig. 18 - Ecrã principal do programa MainTech.

Na barra de superior existem cinco menus: **Adicionar**, **Editar**, **Novo**, **Avançado** e **Modo do Programa**. Dentro deles existem submenus. O menu Adicionar tem quatro submenus: Empresa, Elemento, LIPS e Categoria. O menu Editar tem dois submenus: Empresa e Categoria. O menu Novo tem apenas o submenu Registo. O menu Avançado tem quatro submenus: Verificar Elementos, Consulta Base de Dados, Relatório de actividades e Verificar Garantias. O menu Modo do Programa tem dois submenus: Operador e Gestor.

Na parte inferior do programa, encontram-se as pastas: **Empresa Fabricante**, **Empresa Montagem**, **Registo**, **Manutenção** e **Outras Pesquisas** respectivamente.

7.5.1. Modo do Programa

O menu superior depende do **Modo do Programa** seleccionado. O modo do programa, quando o software é iniciado encontra-se em **Modo Operador**. Quando se abre o programa todos os campos do ecrã principal estão em branco, sendo preenchidos quando o leitor encontrar algum elemento.

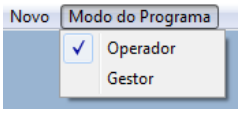


Fig. 19 - Menu superior em modo Operador do programa MainTech.

O **Modo Programa** define o *layout* e as opções visíveis no software, tendo-se duas hipóteses. **Modo Operador** é um modo direccionado para o operador de manutenção e o **Modo Gestor** é um modo direccionado para o gestor de manutenção. Este último possui todas as características do **Modo Operador** mas com a adição de funções específicas ao gestor. Posto isto, apresenta-se o *software* MainTech apenas no **Modo Gestor**.

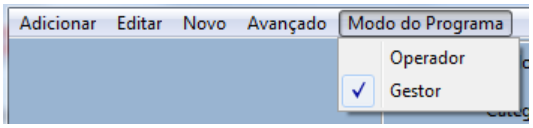


Fig. 20 - Menu superior em modo Gestor do programa MainTech.

7.5.2. LEITURA DE ELEMENTO ATRAVÉS DE RFID

Para ler um elemento, o operador terá de clicar no botão **Ler RFID**, aproximando depois o leitor do elemento a ler. Depois de lido, a antena emite um aviso sonoro e o programa fornecerá todos os dados lidos nesse elemento. Na Fig. **21** - Esquema de funcionamento da Leitura por RFID.Fig. 21 o fluxograma de funcionamento:

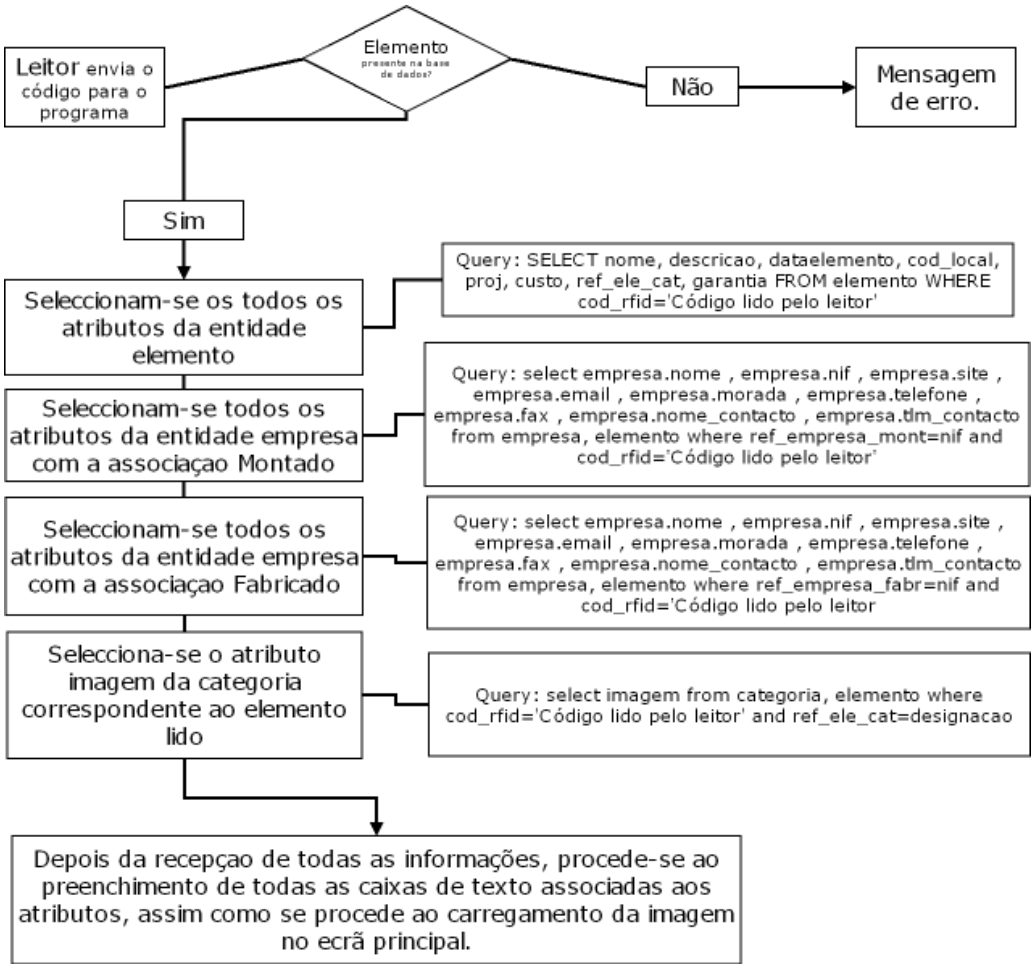


Fig. 21 - Esquema de funcionamento da Leitura por RFID.

Este processo é feito através de 4 *querys* à base de dados: uma em que se obtêm as informações presentes na entidade **Elemento**, outra onde se obtém a imagem e a designação da categoria e as últimas duas para se obterem os dados relativos às informações da empresa fabricante e de montagem. O software grava uma variável global (uma variável que pode ser utilizada em qualquer procedimento) com o número do *tag* lido, com o nome *rfid_codigo_leitura*, para que possa ser utilizado noutros procedimentos. A primeira verificação feita é se o elemento lido está presente na base de dados, caso não esteja, uma mensagem de erro aparece. Após esta verificação são feitas então as 4 consultas, preenchendo então cada campo com a informação da respectiva consulta.

Posto isto, na primeira página estão então os dados do elemento tais como o nome, a categoria que pertence, descrição, data de compra, local, elemento de projecto, custo, garantia e uma fotografia, como se mostra na Fig. 22.

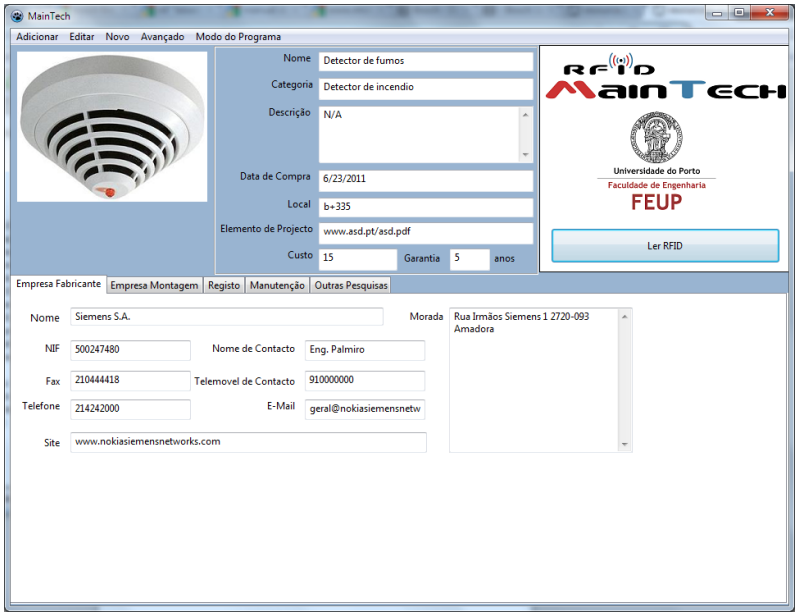


Fig. 22 - Ecrã principal após elemento lido (pasta Empresa Fabricante) do programa MainTech.

Na pasta **Empresa Fabricante** e **Empresa Montagem** encomtram-se as informações relativas a essas empresas. Essas informações são o nome, nif, fax, telefone, site, morada, nome de contacto e telemóvel de contacto.

De salientar que todas as *querys* executadas neste procedimento são de resultado único, isto é, para cada uma apenas obtemos uma linha com as colunas especificadas, dado não existirem dois elementos com o mesmo numero RFID, nem algum elemento pertencente a duas categorias.

7.5.3. Registo

Na pasta **Registo** tem-se uma lista de todos os registos associados a esse elemento, podendo dispor-se essa informação por data, tipo de acção ou estado da acção, como se pode visualizar na Fig. 23. Para isso, basta seleccionar a opção desejada e clicar no botão **Actualizar**.

Para a aquisição dos dados de registo de um elemento, teremos que executar uma *query* que seleccione todos os registos associados ao respectivo número RFID do elemento em questão. Como exemplo, a *query* para os registos associados a um elemento, ordenada por data será:

“SELECT registo.cod_reg, registo.tipo, registo.relatorio, registo.data_reg, registo.estado, registo.observ, registo.custo FROM registo, elemento WHERE registo.ref_ele=cod_rfid and cod_rfid='+rfid_codigo_leitura+' ORDER by registo.data_reg DESC”

Ao contrário dos dados do elemento referidos em 7.5.2., esta consulta à base de dados não será de apenas uma linha pois cada elemento pode possuir inúmeros registos. Posto isto é necessário recorrer a um ciclo Para (FOR) para o preenchimento de uma tabela com os dados recebidos.

Antes de mais é necessário conferir o número de linhas e colunas da tabela a preencher (*TStringGrid*). Este processo é feito recorrendo a duas propriedades do objecto *SQLQuery*: *RecordCount* e *FieldCount*. Que são respectivamente o número de linhas recebidas da *query* efectuada, e o número de informações requisitadas recebidas (neste caso seriam 7: Código, Tipo, Relatório, Data, Estado, Observações e Custo).

Depois disso recorre-se a dois ciclos Para (*for*) para proceder-se ao preenchimento de cada campo da respectiva tabela. O primeiro tem como função percorrer todas as linhas, e o segundo, dentro de cada linha irá percorrer todas as colunas. Em seguida apresenta-se o código usado para o preenchimento de tabelas.

```
for ln := 1 to SQLQuery.RecordCount do
  begin
    SQLQuery.RecNo := ln;
      for col := 0 to SQLQuery.FieldCount - 1 do
    begin
        StringGrid2.Cells[col, ln] := SQLQuery.Fields[col].AsString;
     end;
  end;
```

Após o preenchimento da tabela, procede-se a outro ciclo para que se ajuste o tamanho de cada coluna à maior informação contida dentro dela, isto faz com que seja possível uma leitura integral, independentemente do tamanho da informação contida. Ao contrário do ciclo de preenchimento, este começará por seleccionar uma coluna, percorrendo depois todas as linhas dessa coluna. Na primeira linha grava-se o tamanho do texto contido para uma variável local do procedimento, passando para a segunda linha apenas se gravara o tamanho caso seja maior que o anterior. Deste modo, na linha final da coluna a variável terá o valor do maior texto da coluna, definido então o tamanho dessa coluna e passando para a próxima. Segue-se um excerto do código usado.

```
for col:=0 to SQLQuery.FieldCount -1  do
    begin
    tamanho_maximo:=0;
        for ln:=0 to SQLQuery.RecordCount do
            begin
            tamanho_temp:=(Canvas.TextWidth(StringGrid2.cells[col,ln]));
                if tamanho_temp>tamanho_maximo then begin
                  tamanho_maximo:=tamanho_temp;
                end;
            StringGrid2.ColWidths[col]:=tamanho_maximo+10;
            end ;
    end;
```

No caso da tabela registo (entidade **Registo**) o software irá receber as informações de **Tipo** e **Estado** em valores numéricos. Para uma melhor leitura do operador, o software tem que associar cada um dos códigos à designação correspondente. Ver capítulo 7.3.5. Para isso executam-se simples ciclos "Se" (if) para se proceder-se à alteração: Exemplifica-se com o código utilizado:

```
for ln := 1 to SQLQuery.RecordCount do begin          //substituir códigos tipo por nomes
    if StringGrid2.Cells[1,ln]='1' then begin StringGrid2.Cells[1,ln]:='Limpeza'; end
    else if StringGrid2.Cells[1,ln]='2' then begin StringGrid2.Cells[1,ln]:='Inspecção'; end
    else if StringGrid2.Cells[1,ln]='3' then begin StringGrid2.Cells[1,ln]:='Pró-Acção'; end
    else if StringGrid2.Cells[1,ln]='4' then begin StringGrid2.Cells[1,ln]:='Substituição'; end
    else if StringGrid2.Cells[1,ln]='5' then begin StringGrid2.Cells[1,ln]:='Correcção'; end
    end;
 for ln := 1 to SQLQuery.RecordCount do begin          //substituir códigos estado por nomes
    if StringGrid2.Cells[4,ln]='1' then begin StringGrid2.Cells[4,ln]:='Programado'; end
    else if StringGrid2.Cells[4,ln]='2' then begin StringGrid2.Cells[4,ln]:='Em Execução'; end
    else if StringGrid2.Cells[4,ln]='3' then begin StringGrid2.Cells[4,ln]:='Executado'; end
    end;
```

Se seleccionado outra disposição para a informação, a *query* irá mudar para que se obtenham as informações dispostas da maneira pretendida, correndo depois os mesmo ciclos de preenchimento e ajuste do tamanho assim como a alteração de códigos para informação legível pelo utilizador.

***Query* de registos dispostos por tipo de acção: "**SELECT registo.cod_reg, registo.tipo, registo.relatorio, registo.data_reg, registo.estado, registo.observ, registo.custo FROM registo, elemento WHERE registo.ref_ele=cod_rfid and cod_rfid='+rfid_codigo_leitura+' ORDER by registo.tipo"

***Query* de registos dispostos por estado da acção:**"SELECT registo.cod_reg, registo.tipo, registo.relatorio, registo.data_reg, registo.estado, registo.observ, registo.custo FROM registo, elemento WHERE registo.ref_ele=cod_rfid and cod_rfid='+rfid_codigo_leitura+' ORDER by registo.estado"

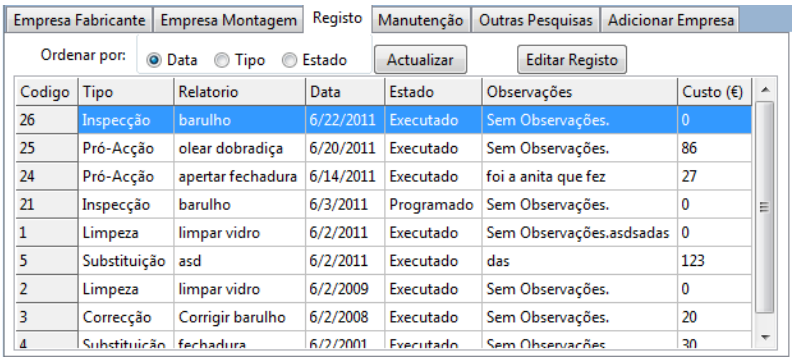


Fig. 23 - Pasta de Registo do programa MainTech.

7.5.4. Edição de Registos

Tem-se também possibilidade de editar registos, bastando para isso seleccionar o registo a editar e clicar no botão **Editar Registo**. Com esta ordem abre-se uma nova pasta, Fig. 24, onde se edita a informação desse registo, podendo alterar os seus dados assim como o estado da tarefa.

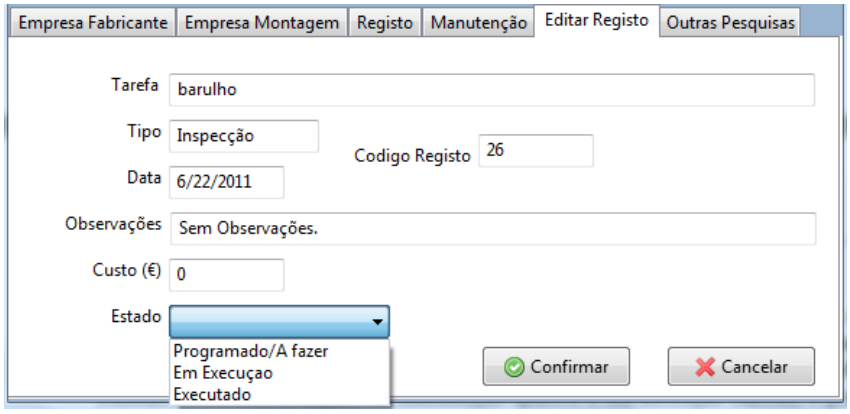


Fig. 24 - Pasta Editar Registos do programa MainTech.

Neste processo de edição de registos toda a informação contida na linha seleccionada da tabela dos registos, irá passar automaticamente para as caixas de texto respectivas presentes na pasta **Editar**

Registo. Este processo é feito pela atribuição da propriedade *text* dos objectos *TEdit* presentes nesta pasta do seguinte modo:

```
edit_reg_tarefa.text:=StringGrid2.Cells[2,StringGrid2.Row]; //Copiar textos da grelha para as TEdit
edit_reg_cod_reg.text:=StringGrid2.Cells[0,StringGrid2.Row];
edit_reg_tipo.text:=StringGrid2.Cells[1,StringGrid2.Row];
edit_reg_data.text:=StringGrid2.Cells[3,StringGrid2.Row];
edit_reg_observ.text:=StringGrid2.Cells[5,StringGrid2.Row];
edit_reg_custo.text:=StringGrid2.Cells[6,StringGrid2.Row];
TabSheet7.Visible := true;                    // Fazer aparecer a pasta Editar Registo
TabSheet7.TabVisible := true;
TabSheet7.Enabled:=true;
TabSheet7.Show;
```

De salientar que independentemente da pasta aberta é possível aceder às propriedades de objectos noutras pastas. Para a alteração ou criação de informações na base de dados, este processo é feito de maneira diferente das *querys*. Quando se altera uma informação do registo e se clica em Confirmar (*TBitBtn*) é enviado para a base de dados o que se chama de *Work* (trabalho). Deste modo o *work* de alteração do registo será:

```
UPDATE registo SET estado = '+estado+' , observ = '"+edit_reg_observ.text+'", custo = '"+edit_reg_custo.text+'" WHERE cod_reg = '"+edit_reg_cod_reg.text+'""
```

Este comando faz com que se actualizem os campos Estado, Observações e Custo do registo específico. O *work* serve tanto para editar como para adicionar dados à base de dados.

7.5.5. Manutenção

Na pasta **Manutenção**, aparecem todos os processos de manutenção previstos para esse elemento. Esta lista de manutenções poderá ser organizada da maneira que o operador necessitar, aparecendo somente os dados de manutenção de limpezas, inspecções, pró-acções e substituições ou então listar todas as operações, bastando para isso clicar num dos botões **Limpeza**, **Inspecção**, **Pró-Acção**, **Substituição** ou **Todas** respectivamente. Estes botões estão agrupados num quadro com nome **Periódicas**. Na lista das acções de manutenção aparece a sua descrição (tarefa), o procedimento, o responsável, a periodicidade e o custo previsto associado a estas.

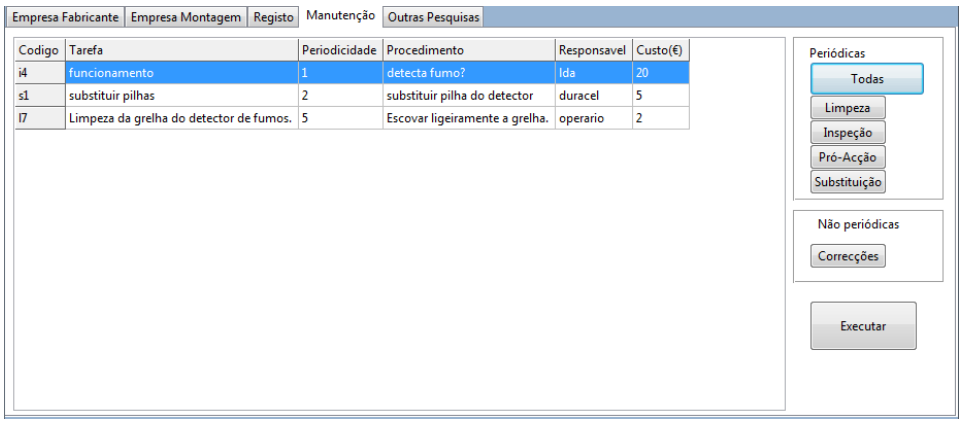


Fig. 25 - Pasta Manutenção com acções periódicas do programa MainTech.

Ainda nesta página está presente uma lista de correcções associadas a este elemento, diferindo das manutenções periódicas por não terem uma periodicidade definida. Para aceder a esta lista clica-se no botão **Correcções**, no quadrado de **Não Periódicas**.

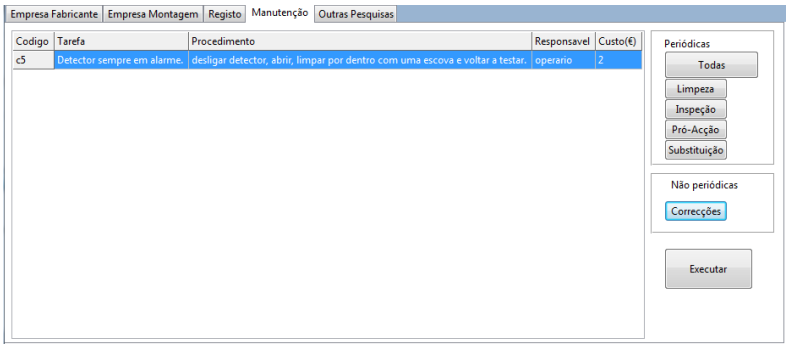


Fig. 26 - Pasta Manutenção acções não periódicas do programa MainTech.

Cada botão terá a sua *query* respectiva, que visa obter uma tabela com os valores Código, Tarefa, Procedimento, Responsável, Periodicidade e Custo das acções de manutenção associadas à categoria do elemento lido. De seguida, expõem-se as *querys* associadas aos botões:

Todas: SELECT lips.cod_lips, lips.tarefa , lips.periodicidade , lips.procedimento, lips.responsavel,lips.custo FROM lips INNER JOIN possui ON possui.ref_lips=lips.cod_lips WHERE possui.ref_cat='"+categoria_leitura+'" AND cod_lips NOT LIKE "c%" ORDER BY lips.periodicidade

Limpeza: SELECT lips.cod_lips, lips.tarefa , lips.periodicidade , lips.procedimento, lips.responsavel,lips.custo FROM lips INNER JOIN possui ON possui.ref_lips=lips.cod_lips WHERE possui.ref_cat='"+categoria_leitura+'" AND cod_lips ILIKE "l%" ORDER BY lips.periodicidade

Inspecção: SELECT lips.cod_lips, lips.tarefa , lips.periodicidade , lips.procedimento, lips.responsavel,lips.custo FROM lips INNER JOIN possui ON possui.ref_lips=lips.cod_lips WHERE possui.ref_cat='"+categoria_leitura+'" AND cod_lips ILIKE "i%" ORDER BY lips.periodicidade

Pró-Acção: SELECT lips.cod_lips, lips.tarefa , lips.periodicidade , lips.procedimento, lips.responsavel,lips.custo FROM lips INNER JOIN possui ON possui.ref_lips=lips.cod_lips WHERE possui.ref_cat='"+categoria_leitura+'" AND cod_lips ILIKE "p%" ORDER BY lips.periodicidade

Substituição: SELECT lips.cod_lips, lips.tarefa , lips.periodicidade , lips.procedimento, lips.responsavel,lips.custo FROM lips INNER JOIN possui ON possui.ref_lips=lips.cod_lips WHERE possui.ref_cat='"+categoria_leitura+'" AND cod_lips ILIKE "s%" ORDER BY lips.periodicidade

Correções: SELECT lips.cod_lips, lips.tarefa , lips.procedimento, lips.responsavel, lips.custo FROM lips INNER JOIN possui ON possui.ref_lips=lips.cod_lips WHERE possui.ref_cat='"+categoria_leitura+'" AND cod_lips ILIKE "c%" ORDER BY lips.tarefa

O preenchimento e ajuste do tamanho das colunas referidos no capítulo 7.5.3 são utilizados de maneira idêntica. Quando o operador pretender executar uma destas tarefas, basta seleccionar uma opção e clicar no botão **Executar** que o levará a outra pasta designada por **Acção**.

7.5.6. Acção

Nesta pasta aparecem as informações sobre a tarefa a executar, tendo os campos **Tarefa**, **Tipo** e **Data** preenchidos automaticamente, não tendo o operador a possibilidade de os alterar. O campo **Custo**

também já estará preenchido com a informação do custo previsto da manutenção, mas o operador tem a possibilidade de o alterar.

Este processo é efectuado em ambiente de execução, copiando a informação da linha escolhida da tabela presente na pasta **Manutenção** quando se clica no botão **Executar**, da seguinte forma:

```
//Obtém-se a data actual para preenchimento do campo data
datetimetostring(data, 'yyyy-mm-dd', now);
accao_data.Text:=data;
accao_tarefa.text:=StringGrid1.Cells[1,StringGrid1.Row];
//Atribuem-se os valores da grelha da pasta Manutenção para os campos de texto na pasta Acção
accao_tipo.text:=StringGrid1.Cells[0,StringGrid1.row];
accao_custo.Text:=StringGrid1.Cells[5,StringGrid1.Row];
MemoAccaoProcedimento.Append(StringGrid1.Cells[3,StringGrid1.Row]);
//Neste caso transformam-se as designações do tipo de manutenção no seu código correspondente
if accao_tipo.text[1]='i' then begin accao_tipo.text:='Inspecção'; accao_tipo.tag:=2; end
else if accao_tipo.text[1]='p' then begin accao_tipo.text:='Pró-Acção'; accao_tipo.tag:=3; end
else if accao_tipo.text[1]='l' then begin accao_tipo.text:='Limpeza'; accao_tipo.tag:=1; end
else if accao_tipo.text[1]='s' then begin accao_tipo.text:='Substituição'; accao_tipo.tag:=4; end
else if accao_tipo.text[1]='c' then begin accao_tipo.text:='Correcção'; accao_tipo.tag:=5; end;
//Comandos para fazer surgir a pasta Acção
tabsheet5.TabVisible:=true;
tabsheet5.Visible:=true;
TabSheet5.Enabled:=true;
TabSheet5.Show;
```

Para o operador confirmar a acção terá de preencher o campo **Estado da tarefa** para uma das opções **Programado/A fazer**, **Em execução** e **Executado**. Pode também, se necessitar, preencher o campo **Observações** e de seguida, clicar no botão **Confirmar**, o qual lança um aviso se efectivamente quer registar a acção. De referir que o procedimento da tarefa se encontra disponível para leitura do operador na pasta **Acção**, deste modo o operador tem toda a informação do procedimento da acção no momento em que a está a executar. Em qualquer altura o operador pode cancelar a tarefa.

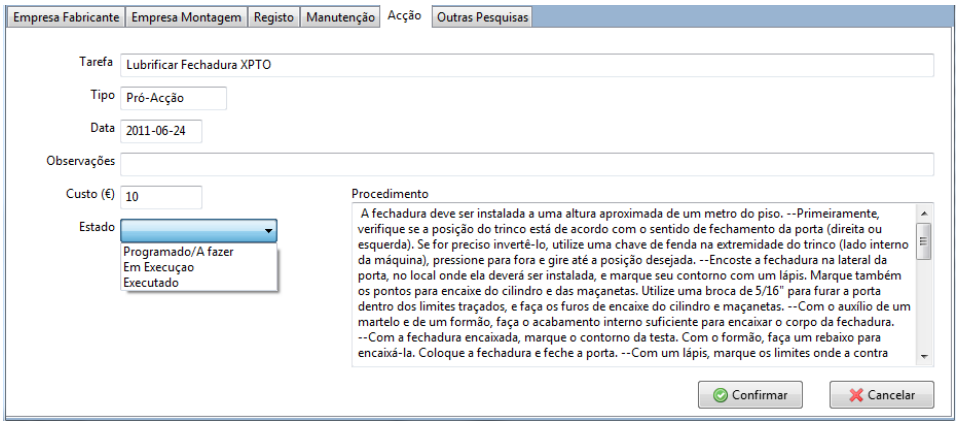


Fig. 27 - Pasta Acção para um exemplo de correcção do programa MainTech.

Após o utilizador clicar no botão **Confirmar**, o software irá gerar o seguinte *Work* para que esta acção entre na lista de registos associados ao elemento:

INSERT INTO registo (custo, estado, tipo, observ, relatorio, data_reg, ref_ele) VALUES ('+accao_custo.text+','+estado+','"+inttostr(accao_tipo.tag)+"','"+accao_observ.Text+"','"+accao_tarefa .Text+"','"+accao_data.text+"','+rfid_codigo_leitura+')

Não é necessária a atribuição de um número para o atributo *cod_reg* pois este é gerado automaticamente pela base de dados devido ao seu tipo de dados ser *Serial*.

7.5.7. Novo Registo

É também possível adicionar um registo de manutenção que não esteja presente nos processos de manutenção predefinidos. Tem-se a necessidade de tal pois nem todas as intervenções estão previstas nos processos de manutenção. Para isso, basta clicar no menu superior **Novo** e de seguida clicar em **Registo.**

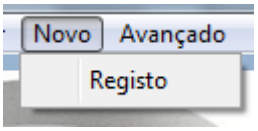


Fig. 28 - Menu Novo do programa MainTech.

Quando se clica em **Registo**, uma nova pasta surge com o nome **Novo Registo**. Esta pasta, apesar de idêntica à pasta **Acção**, difere na inserção dos dados. Todos os campos de texto estão vazios excepto a data que é predefinida com a data actual e o campo **Tipo** aparece em estilo de *combobox* para seleccionar entre **Limpeza**, **Inspecção**, **Pró-Acção**, **Substituição** ou **Correcção** como se pode verificar na Fig. 29.

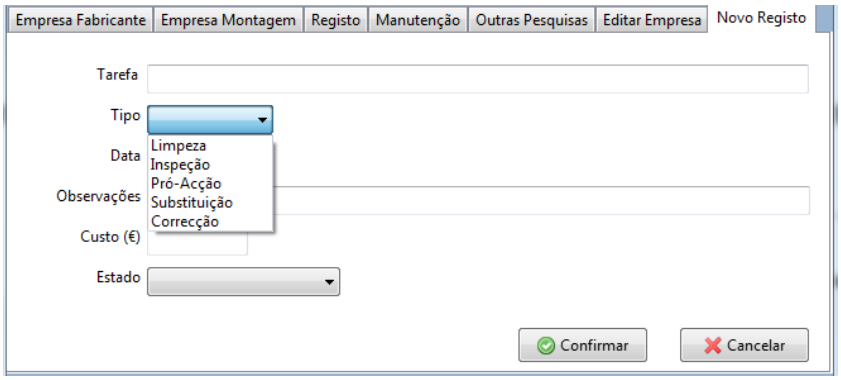


Fig. 29 - Pasta Novo Registo do programa MainTech.

O *work* gerado por este novo registo será idêntico a 7.5.6.

7.5.8. Outras Pesquisas

Na pasta **Outras Pesquisas** tem-se a possibilidade de fazer outras pesquisas relacionadas ou não com o elemento lido. São possíveis as pesquisas: **Outros elementos no mesmo local**, **Elementos com acções programadas**, **Elementos com acções em execução** e **Todos os elementos da mesma categoria**. Destas pesquisas, apenas a que se denomina "Outros elementos no mesmo local" interessa ao operador pois, com recurso à mesma, é possível executar operações de manutenção a elementos que

não estão identificados por RFID e que estão presentes na base de dados. Todas as outras interessam apenas ao gestor de manutenção, que consegue de maneira rápida visualizar todos os elementos que têm acções programadas ou em execução. Os campos de outras pesquisas actualizam com a escolha do **Modo do Programa**. A Fig. 31 mostra o aspecto desta pasta no modo de operador.

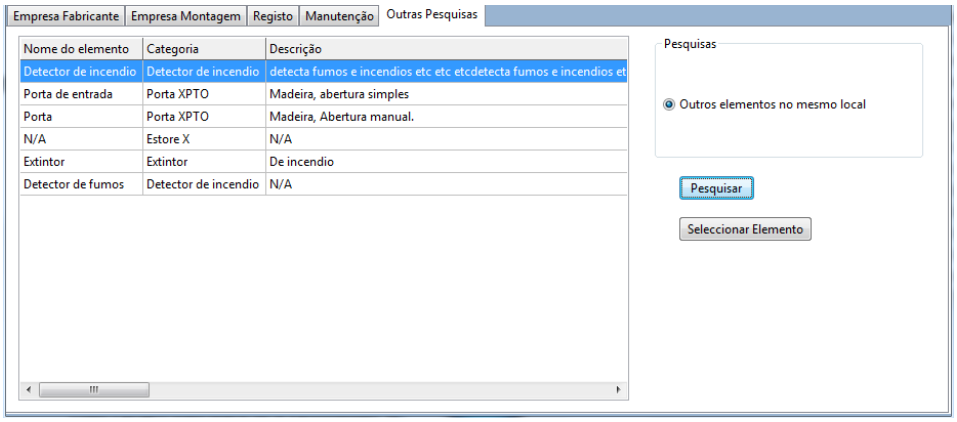


Fig. 30 - Pasta Outras Pesquisas modo Operador do programa MainTech.

Para seleccionar algum dos elementos do resultado da pesquisa, basta seleccionar da lista e clicar no botão **Seleccionar Elemento**. Quando pressionado, no ecrã principal, o elemento lido vai ser alterado pelo elemento seleccionado, e todas as pastas actualizadas para o elemento actual. A Fig. 31 mostra esta pasta no modo de gestor.

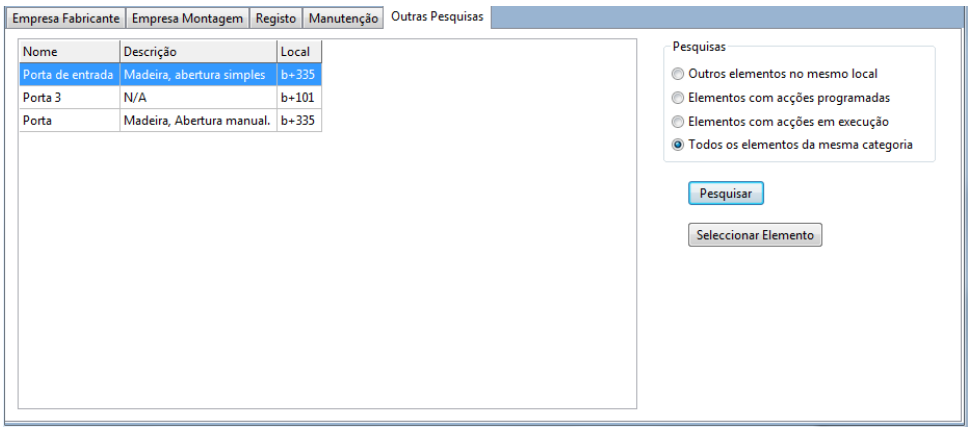


Fig. 31 - Pasta Outras Pesquisas modo Gestor do programa MainTech.

Indica-se agora a *query* respectiva de cada pesquisa:

Outro elementos no mesmo local: SELECT nome AS "Nome do elemento", ref_ele_cat AS "Categoria" , descricao AS "Descrição" , cod_rfid FROM elemento WHERE cod_local="'+memo4.text+'"

Elementos com acções programadas: SELECT elemento.nome AS "Elemento",ref_ele_cat AS "Categoria",cod_local AS "Local",data_reg AS "Data",relatorio AS "Relatório",observ AS "Observações",cod_rfid FROM elemento, registo WHERE registo.estado=1 AND registo.ref_ele=cod_rfid

Elementos com acções em execução: SELECT elemento.nome AS "Elemento",ref_ele_cat AS "Categoria",cod_local AS "Local",data_reg AS "Data",relatorio AS "Relatório",observ AS "Observações",cod_rfid FROM elemento, registo WHERE registo.estado=2 AND registo.ref_ele=cod_rfid
Todos os elementos da mesma categoria: SELECT nome AS "Nome",descricao AS "Descrição",cod_local AS "Local", cod_rfid FROM elemento WHERE ref_ele_cat='"+memo7.text+""

Depois do utilizador seleccionar um dos elementos, o software terá de correr o processo descrito em 7.5.2, mas com o código RFID do elemento seleccionado no resultado da pesquisa. De realçar que o código RFID não se encontra disponível ao utilizador, estando escondido numa coluna da tabela com largura zero.

7.5.9. Menu Adicionar

Explicam-se agora as características do software específicas do gestor. Começando pelo menu superior, a sua primeira opção é Adicionar. Dentro do menu **Adicionar**, tem os seguintes submenus: **Empresa**, **Elemento**, **LIPS** e **Categoria**.

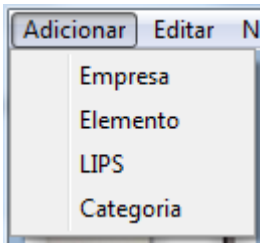


Fig. 32 - Menu superior Adicionar e submenus do programa MainTech..

Como o próprio nome indica, permite-se a possibilidade de adicionar empresas, elementos, processos de manutenção e categorias à base de dados.

7.5.9.1. Empresa

Para adicionar uma empresa segue-se o trajecto **Adicionar** e, de seguida, **Empresa**. Irá então aparecer uma nova pasta com o nome de **Adicionar Empresa**, onde se inserem os dados dessa nova empresa (nome, nif, fax, telefone, site, morada, nome de contacto e telemóvel de contacto). Os campos Nome e NIF são obrigatórios sendo todos os outros opcionais.

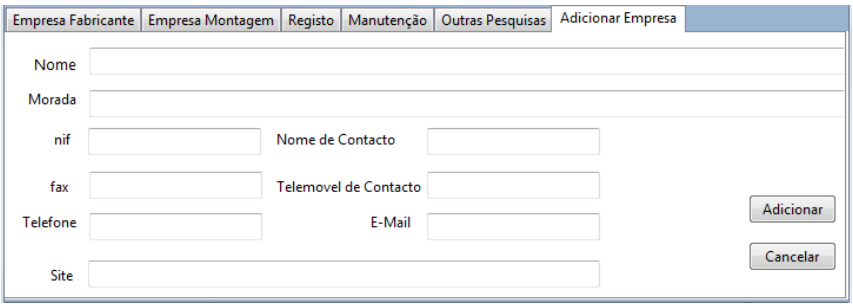


Fig. 33 - Pasta Adicionar Empresa do programa MainTech..

O software lança uma mensagem de erro caso estes não estejam preenchidos e, caso os outros campos não estejam preenchidos, serão automaticamente preenchidos com o texto “N/A” (Não Aplicável) ou com um algarismo “0” no caso de a informação ser numérica (Telefone, Fax e telemóvel de contacto).

Depois de clicar no botão **Adicionar**, o software corre um ciclo que em primeiro lugar verifica se os campos Nome e Nif estão preenchidos, correndo depois outro ciclo de verificação dos restantes campos opcionais. Só depois das verificações é que o software cria o *work* de inserção dos dados na entidade **Empresa**. O organograma seguinte demonstra simplificadamente os passos executados pelo software.

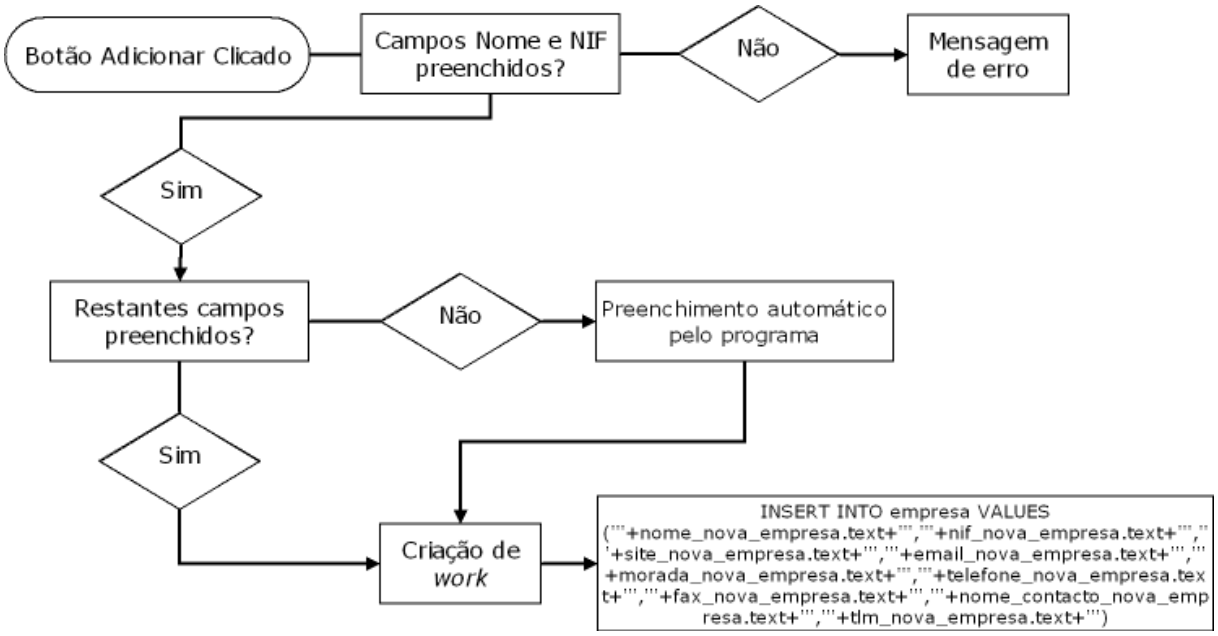


Fig. 34 - Organograma da criação de empresas.

7.5.9.2. Categoria

Para adicionar uma Categoria, clica-se em **Adicionar** e depois **Categoria**. Abrir-se-á uma nova pasta onde se preenche o nome da categoria e onde terá um botão para adicionar a imagem da respectiva categoria, imagem essa que terá de constar no disco do computador para poder ser carregada. O objecto *TOpenPictureDialog* é utilizado no processo de carregamento da imagem.

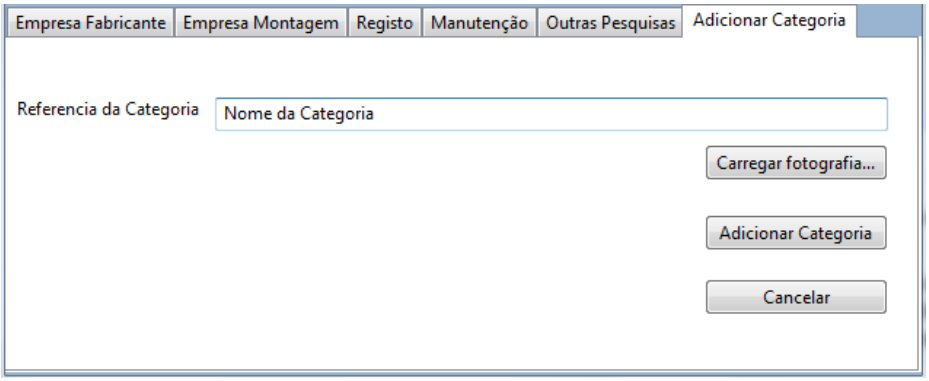


Fig. 35 - Pasta Adicionar Categoria do programa MainTech..

As imagens de cada categoria serão carregadas para o directório *images* dentro da pasta da aplicação e poderão ter qualquer nome como assim como a maior parte dos formatos utilizados actualmente (jpg, bmp, png, etc). O código terá o seguinte aspecto:

ImageAddCat.Picture.SaveToFile('images\'+ExtractFileName(OpenPictureDialog1.FileName));
Na entidade Categoria da base de dados, o atributo Imagem é gravado apenas com o nome do ficheiro da imagem, isto porque, na leitura de um elemento corre-se uma *querie* para saber qual o nome da imagem a carregar e o software irá carregar a imagem do directório *images*.

Após preenchimento do nome da categoria e carregamento da fotografia, clica-se em **Adicionar** e a categoria passará a ser integrada na base de dados. O *work* gerado pelo programa tem o seguinte aspecto:

INSERT INTO categoria (designacao, imagem) VALUES ('"+Edit_design_Categoria.text+'" , '"+ExtractFileName(OpenPictureDialog1.FileName)+'")

7.5.9.3. LIPS

Para adicionar processos de manutenção, clica-se no menu **Adicionar** e no submenu **LIPS**. Aparece um nova pasta com o nome **Adicionar LIPS**.

Em primeiro lugar, selecciona-se o tipo de tarefa a adicionar, podendo ela ser uma limpeza, inspecção, pró-acção, substituição ou correcção. Esta selecção é feita através de uma *combobox* como se pode ver pela Fig. 36. Depois existem três campos de texto onde se define a tarefa, o procedimento, custo previsto e o responsável por ela. Existe também um campo de periodicidade que só está apto a ser alterado no caso de se ter escolhido uma acção com periodicidade, quando seleccionado **Correcções** o campo fica inactivo com o valor zero.

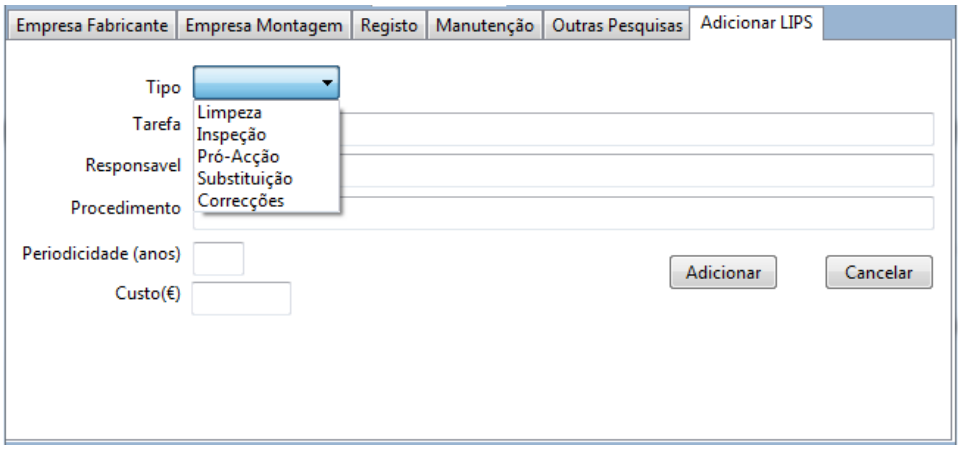


Fig. 36 - Pasta Adicionar LIPS do programa MainTech..

Dada a contagem definida para a *primary key* (cof_lips) da entidade LIPS em 7.3.5.5 o software terá de proceder a um ciclo para determinar qual o novo código. Para isto, em primeiro lugar o software irá verificar qual o tipo de manutenção a adicionar. Depois o software irá executar uma *query* para verificar qual o número do ultimo processo de manutenção desse tipo. O organograma seguinte explica simplificadamente o processo de obtenção do novo código do processo de manutenção.

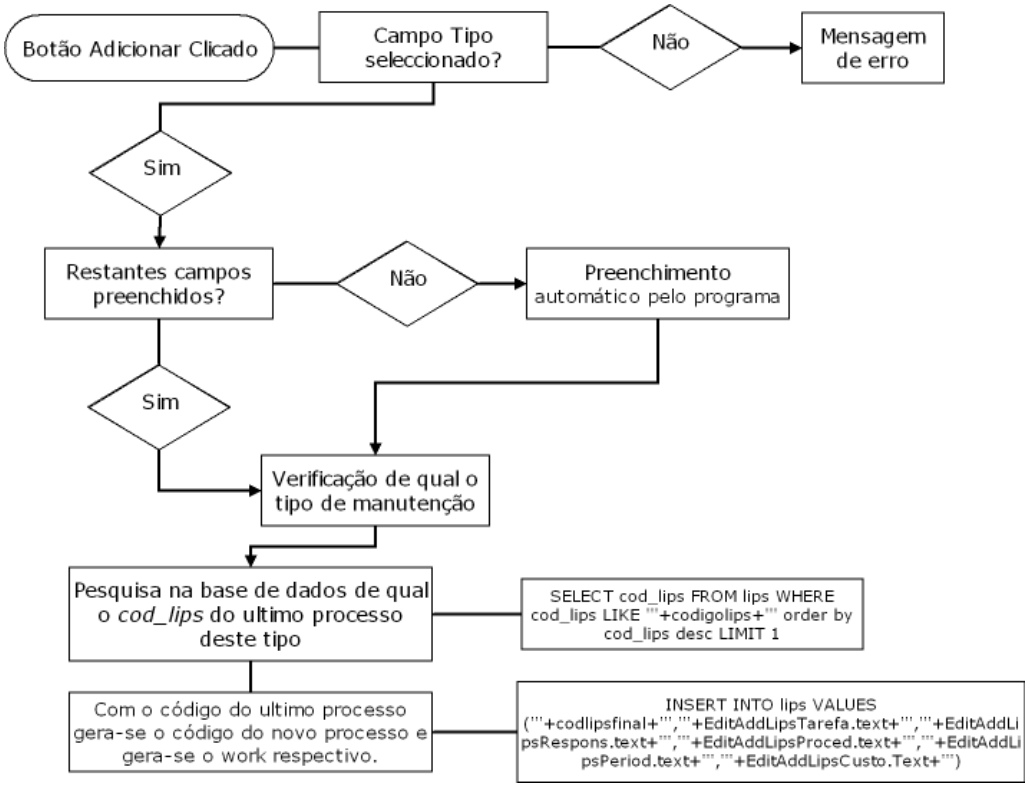


Fig. 37 - Organograma de obtenção de novo *cod_lips*.

O procedimento descrito pode ser também exemplificada em formato de código:

```
// Verificação de preenchimento do tipo da manutenção e respectiva mensagem de erro
if (ComboBoxTipoLIPS.ItemIndex=-1) or (EditAddLipsTarefa.text='') then begin
  showmessage('Falta descrição da tarefa ou tipo.'); end
  else begin butao_pressionado:=MessageDlg('Confirmar nova tarefa?',mtCustom,[mbYes,mbNo], 0);
  if butao_pressionado=mryes then begin
// Detecção do formato do código do respectivo tipo de manutenção
  if ComboBoxTipoLIPS.ItemIndex=0 then begin codigolips:='l%'; end
  else if ComboBoxTipoLIPS.ItemIndex=1 then begin codigolips:='i%'; end
  else if ComboBoxTipoLIPS.ItemIndex=2 then begin codigolips:='p%'; end
  else if ComboBoxTipoLIPS.ItemIndex=3 then begin codigolips:='s%'; end
  else if ComboBoxTipoLIPS.ItemIndex=4 then begin codigolips:='c%'; end;
// Query para verificar qual o ultimo processo deste tipo na base de dados
  SQLQuery.SQL.Text:='SELECT cod_lips FROM lips where cod_lips Like '''+codigolips+''' order by
cod_lips desc LIMIT 1';
  PQConnection.Open;
  SQLQuery.Open;
  SQLQuery.First;
  SQLQuery.Last;
  SQLQuery.RecNo := 1;
  codigolips_numero:=SQLQuery.Fields[0].AsString;
  SQLTransaction.Active := False;
  SQLQuery.Close;
  PQConnection.close;
// Processo de criação do novo código
  Delete(codigolips_numero,1,1);
  codlipsfinal:=codigolips[1]+inttostr(strtoint(codigolips_numero)+1);
// Criação do work da adição da respectiva manutenção à entidade LIPS
```

```
PQConnection.Open;
  PQConnection.ExecuteDirect('Begin Work;');
  sql_add_lips:='insert into lips values ('''+codlipsfinal+''','''+EditAddLipsTarefa.text+''','''+EditAddLipsRespons.text+''','''+EditAddLipsProced.text+''','''+EditAddLipsPeriod.text+''','''+EditAddLipsCusto.Text+''')';
  PQConnection.ExecuteDirect(sql_add_lips);
  PQConnection.ExecuteDirect('Commit Work;');
  PQConnection.Close;
  showmessage('Tarefa adicionada com sucesso.');
// Fecho da pasta Adicionar LIPS
 TabSheetAddLips.TabVisible:=false;
  TabSheetAddLips.Enabled:=false;
  EditAddLipsPeriod.clear;
  EditAddLipsProced.clear;
  EditAddLipsRespons.clear;
  EditAddLipsTarefa.clear;
  ComboBoxTipoLIPS.ItemIndex:=-1;
  end;
```

De salientar que se adicionam processos de manutenção sem estarem ligados a nenhuma categoria. Deste modo, pode utilizar-se um processo para várias categorias. Para ligar uma categoria aos processos de manutenção, clica-se no menu **Editar** e de seguida em **Categoria**. Uma nova pasta com o nome **Editar Categoria** surge.

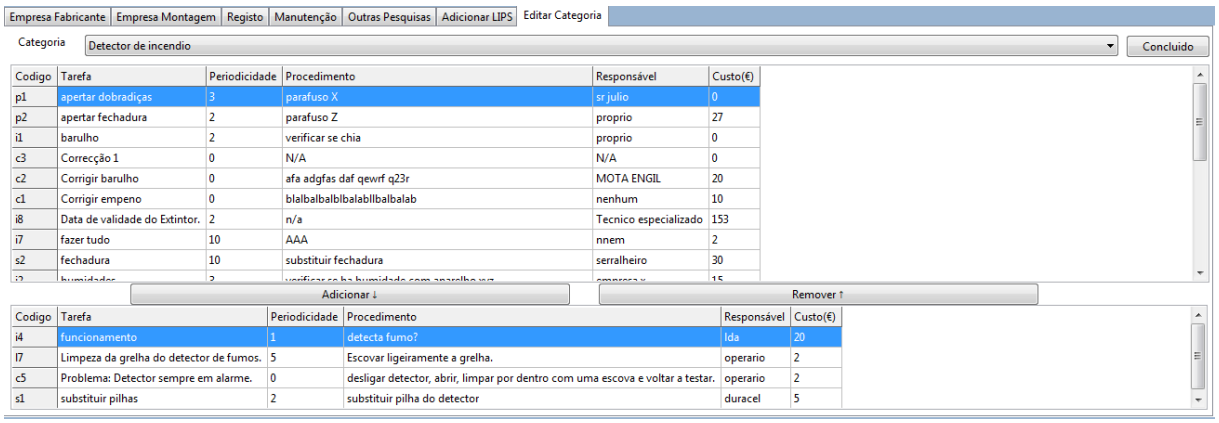


Fig. 38 - Pasta Editar Categoria do programa MainTech.

Dentro dessa pasta de início aparece a lista em formato de *combobox* de todas as categorias presentes na base de dados. O processo de preenchimento de uma *combobox* com dados da base de dados é feito através de dois processos: primeiro através de uma *querie* à base de dados requisitam-se os valores necessários; em segundo lugar cria-se um ciclo que preencha os campos dessa *combobox*.

O código para preenchimento da *combobox* da pasta **Editar Categoria** com as designações das categorias presentes na base de dados é o seguinte:

```
// Limpar os dados que possam estar na combobox
ComboBoxEditCat.Clear;
// Pesquisar na base de dados as designações de todas as categorias com a seguinte query
 SQLQuery.sql.text:='select designacao from categoria order by designacao';
 PQConnection.Open;
 SQLQuery.Open;
 SQLQuery.First;
```

```
 SQLQuery.Last;
// Ciclo de preenchimento de cada linha da Combobox
 for i:=1 to SQLQuery.RecordCount do
   begin
     SQLQuery.RecNo := i ;
     ComboBoxEditCat.Items.add(SQLQuery.fields[0].AsString);;
   end;
```

Após seleccionada a categoria pretendida, irão aparecer então duas listas de processos de manutenção. A lista superior contém todos os processos de manutenção que não estão ligados à categoria e a lista inferior contém uma lista de todos os processos de manutenção associados à categoria. O seguinte organograma explica a obtenção destas tabelas.

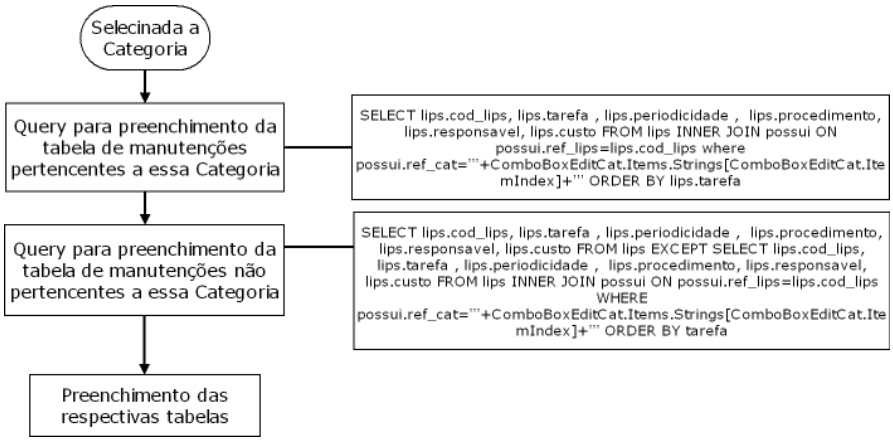


Fig. 39 - Organograma de preenchimento de tabelas na pasta Editar LIPS

Para adicionar ou remover processos de manutenção, basta optar pelo processo numa das tabelas e clicar em **Adicionar** ou **Remover** conforme se pode ver na Fig. 38. O seguinte diagrama descreve o processo de adição e remoção de processos de manutenção de uma categoria, assim como os *works* respectivos.

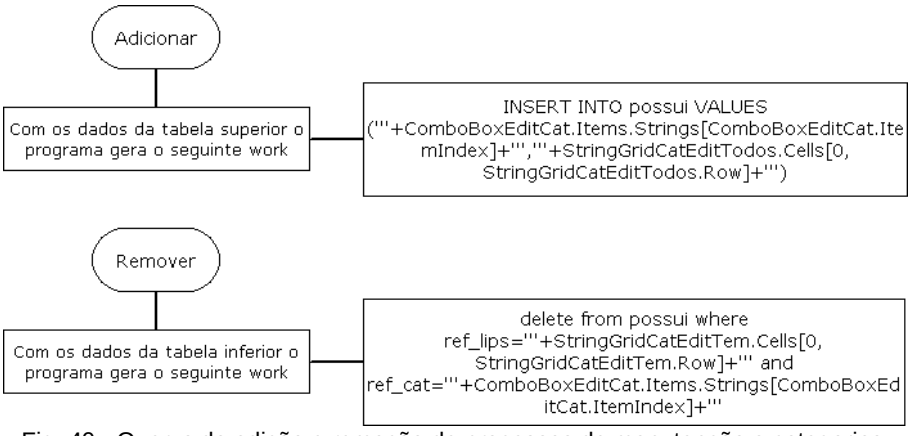


Fig. 40 - Querys da adição e remoção de processos de manutenção a categorias.

7.5.9.4. Elemento

Para adicionar um elemento, clica-se no menu **Adicionar** e no submenu **Elemento**. Uma nova pasta com o nome **Adicionar Elemento** surge. Nesta caixa podem-se preencher os vários dados ligados ao elemento como nome, descrição, data de compra, local, elemento de projecto, custo e garantia.

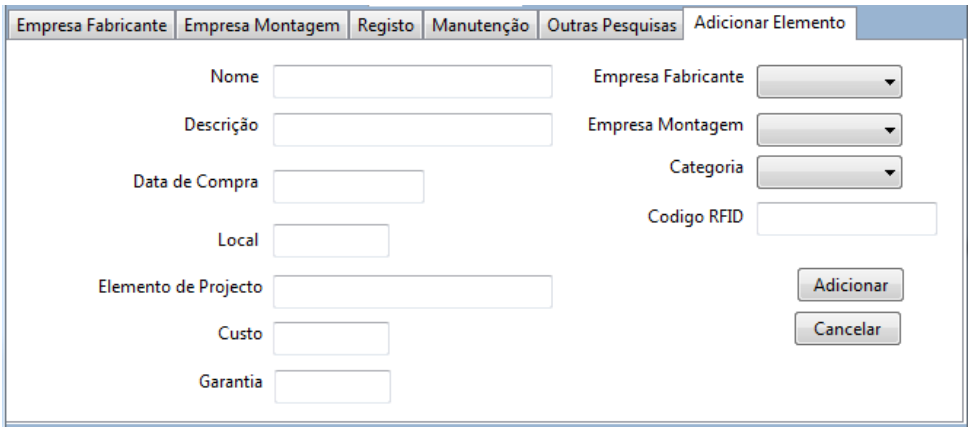


Fig. 41 - Pasta Adicionar Elemento do programa MainTech.

A associação do elemento à empresa fabricante e à empresa de montagem é feita através de uma lista *combobox*, onde aparecem todas as empresas presentes na base de dados e a sua categoria é seleccionada do mesmo modo, aparecendo a lista de todas as categorias em formato *combobox*.

O processo de preenchimento destas *combobox* é idêntico ao referido no capítulo anterior mas, claro, com os *query*s dos dados respectivos.

Para finalizar a operação, clica-se no campo do **Código RFID** e aproximamos a antena do elemento identificado por RFID para que o campo possa ser preenchido pelo respectivo código contido no tag já presente no elemento.

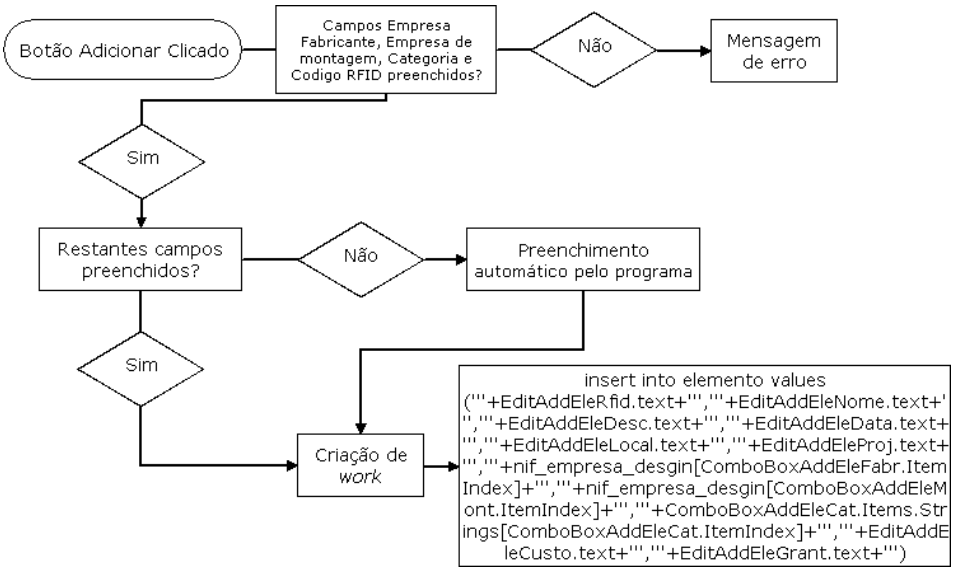


Fig. 42 - Organograma e *query* da pasta Adicionar Elemento.

Caso algum contacto ou morada de uma empresa seja alterado é também possível actualizar essas informações na base de dados. Para isso, clica-se no menu **Editar** e, de seguida, em **Empresa**. Uma nova pasta com o nome de **Editar Empresa** surge, sendo necessário escolher numa caixa estilo *combobox* a empresa que se pretende alterar.

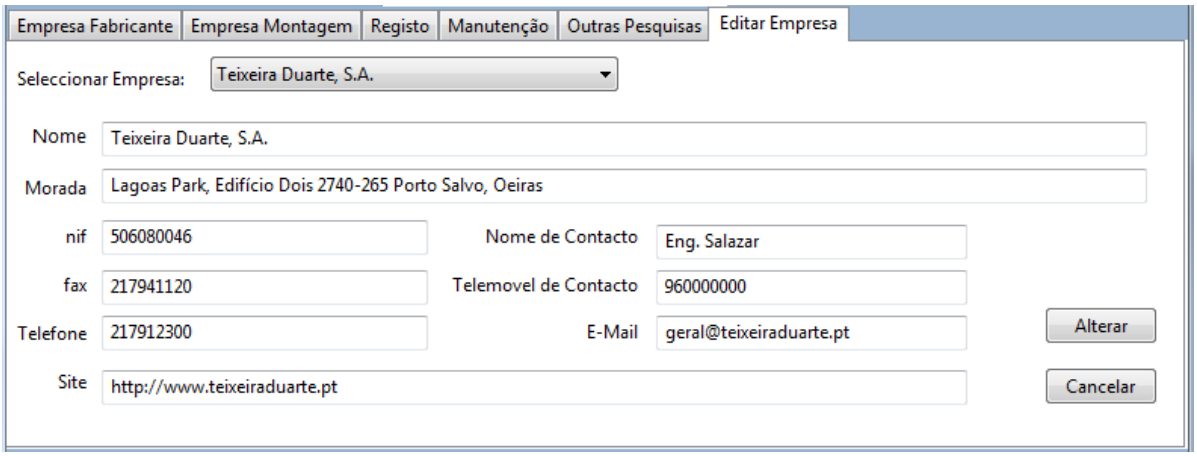


Fig. 43 - Pasta Editar Empresa do programa MainTech.

Depois de alterada a informação pretendida, clica-se em **Alterar** para alterar os valores da empresa na base de dados. Pode, em qualquer altura, cancelar se pretendido. O *work* gerado pelo software tem o seguinte aspecto:

UPDATE empresa SET nome='"+Editar_empresa_1.text+'" , site='"+Editar_empresa_3.text+'" , email='"+Editar_empresa_4.text+'" , morada='"+Editar_empresa_5.text+'" , telefone='"+Editar_empresa_6.text+'" , fax='"+Editar_empresa_7.text+'" , nome_contacto='"+Editar_empresa_8.text+'" , tlm_contacto='"+Editar_empresa_9.text+'" WHERE nif='"+Editar_empresa_2.text+'"

7.5.10. MENU AVANÇADO

O menu **Avançado** tem 4 submenus:

- Verificar Elementos
- Consulta à Base de Dados
- Relatório de Actividades
- Garantias

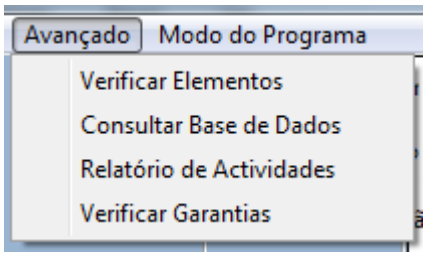


Fig. 44 - Menu Avançado e submenus do programa MainTech.

7.5.10.1. Verificar Elementos

Quando clicado no submenu **Verificar Elementos** surge uma nova pasta com o nome **Verificação de Elementos**.

O objectivo desta pasta é a de verificar se cada elemento já sofreu as acções necessárias relativamente à periodicidade definida para cada acção, num determinado período de tempo determinado pelo utilizador.

Se um elemento tem uma acção com periodicidade de três anos, o programa irá fazer a verificação de qual a data do último registo desse elemento que corresponda à acção de manutenção e verificar se no período de tempo seleccionado este elemento terá que receber, ou não, esse processo de manutenção. No caso de o elemento não ter ainda sofrido nenhuma acção dessa manutenção, o software irá compará-la com a data de compra do elemento.

Esta verificação é bastante importante pois, além de ser um processo bastante moroso de um modo não informatizado, dá a possibilidade ao gestor de criar um plano de trabalhos de um mês ou de um ano. Associado a esta verificação, o software também prevê os custos associados a estas manutenções, dando assim uma estimativa da quantia que irá ser gasta no período de tempo escolhido.

A verificação pode ser feita para todas as categorias ou pode escolher-se uma categoria específica, existindo para isso uma lista no formato *combobox* contendo todas as categorias.

Para seleccionar o intervalo de tempo pretendido existe também uma *combobox* com intervalos desde um mês até a doze meses após a presente data.

Depois de seleccionadas as opções pretendidas clica-se no botão **Verificar Manutenções**.

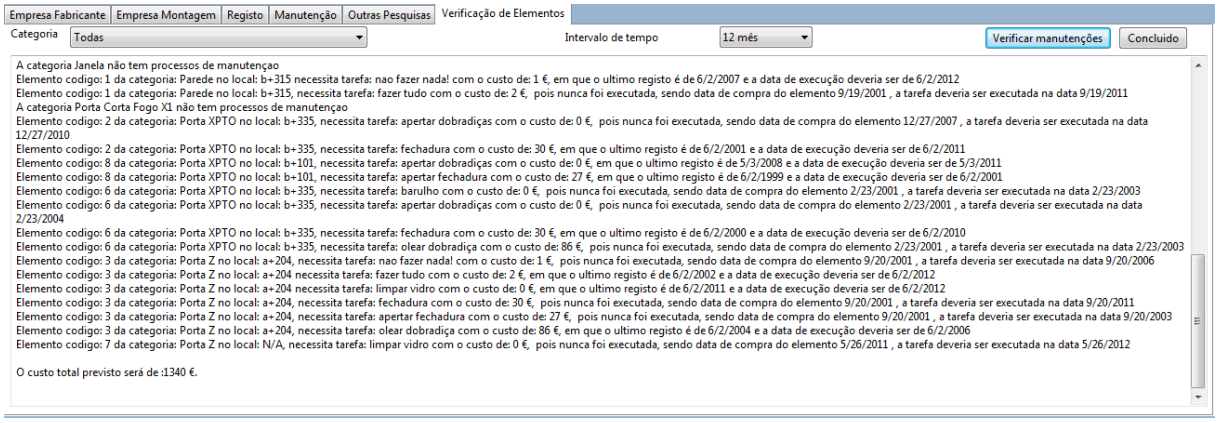


Fig. 45 - Pasta Verificação de Elementos do programa MainTech.

O seguinte diagrama mostra, ainda que de um modo reduzido, o processo executado pelo software. As linhas a tracejado indicam a tabela onde cada acção irá buscar dados para processar a verificação.

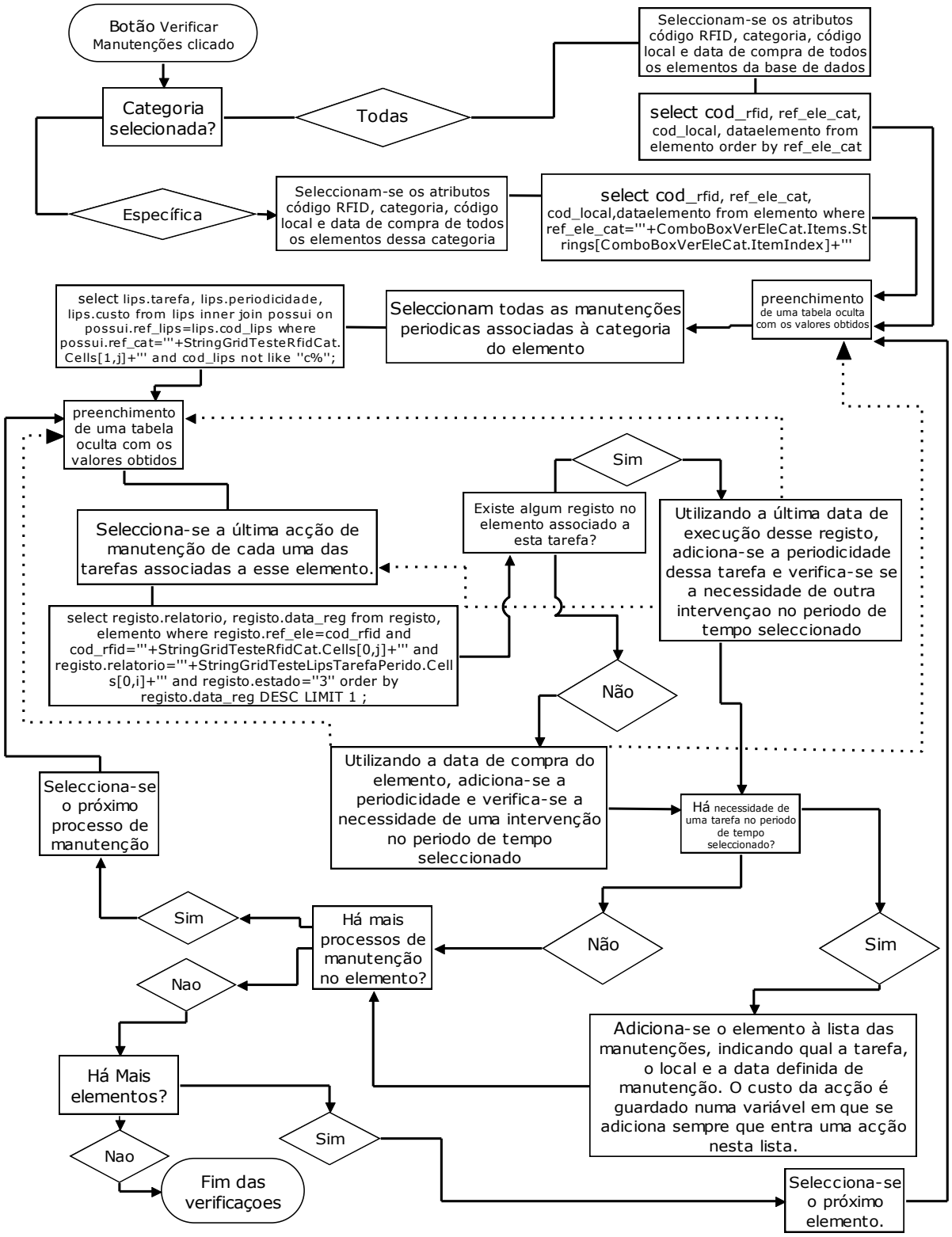


Fig. 46 - Organograma de funcionamento do ciclo de Verificaçoes dos Elementos

O *output* da verificação compreende vários aspectos dependendo da situação. Em primeiro lugar, temos o *output* de uma categoria que não possui processos de manutenção:

"A categoria X não tem processos de manutenção."

Quando se trata de um elemento que não possui nenhum registo de um processo de manutenção específico, o *output* será:

"Elemento código: 16 da categoria: AC Tipo Split no local: a+204, necessita tarefa: Limpar Filtros com o custo de: 10 €, pois nunca foi executada, sendo data de compra do elemento 5/3/2009, a tarefa deveria ser executada na data 5/3/2010"

Quando um elemento já tem algum registo da manutenção específica, o seu *output* será:

"Elemento código: 3 da categoria: Porta Z no local: a+204 necessita tarefa: limpar vidro com o custo de: 2 €, em que o ultimo registo é de 6/2/2011 sendo a sua data de execução prevista de 6/2/2012".

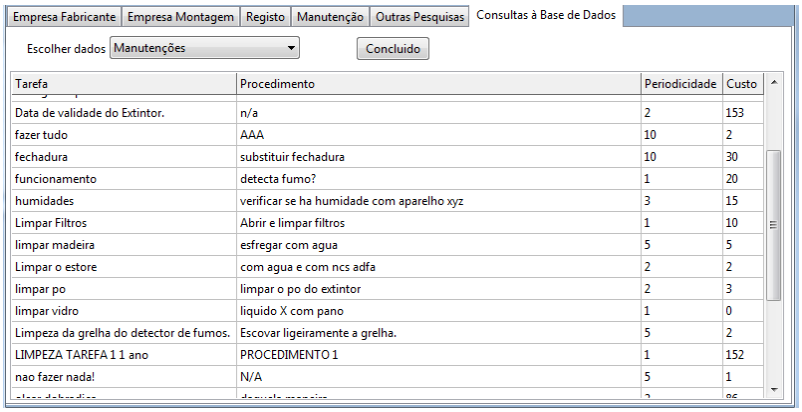
Aliada a esta função, está ainda uma estimativa do orçamento necessário para as manutenções periódicas do edifício. No final do ciclo, encontra-se uma frase com o formato:

" O custo total previsto será de : 11500€ "

Este processo verifica todos os elementos da base de dados e todos os processos de manutenção associados a esses elementos, não tendo algum limite para a quantidade de dados inseridos na base de dados, alterando apenas o seu tempo de execução.

7.5.10.2. Consultar Base de Dados

Quando se clica no menu **Consultar Base de Dados**, uma nova pasta é aberta com o nome **Consultas à Base de Dados**. Esta pasta tem como função inspeccionar o estado das entidades da base de dados. Para isso seleccionam-se os dados que se querem consultar numa *combobox*, pode-se seleccionar: **Elementos**, **Empresas**, **Categoria** e **Manutenções**. Quando seleccionado um, aparecem tabelados os valores de cada uma das entidades.



Tarefa	Procedimento	Periodicidade	Custo
Data de validade do Extintor.	n/a	2	153
fazer tudo	AAA	10	2
fechadura	substituir fechadura	10	30
funcionamento	detecta fumo?	1	20
humidades	verificar se ha humidade com aparelho xyz	3	15
Limpar Filtros	Abrir e limpar filtros	1	10
limpar madeira	esfregar com agua	5	5
Limpar o estore	com agua e com ncs adfa	2	2
limpar po	limpar o po do extintor	2	3
limpar vidro	liquido X com pano	1	0
Limpeza da grelha do detector de fumos.	Escovar ligeiramente a grelha.	5	2
LIMPEZA TAREFA 1 1 ano	PROCEDIMENTO 1	1	152
nao fazer nada!	N/A	5	1

Fig. 47 - Pasta Consulta à Base de Dados do programa MainTech, exemplo de consulta de Manutenções.

Quando é seleccionado o menu **Elementos**, uma tabela dos elementos presentes na base de dados aparece com as informações Código RFID, Nome, Categoria, Local e Data como colunas. Esse processo é feito com a seguinte *query*:

SELECT cod_rfid AS "Código RFID", nome AS "Nome", ref_ele_cat AS "Categoria" , cod_local AS "Local", dataelemento AS "Data" FROM elemento ORDER BY cod_rfid

Em **Empresas**, a tabela é constituída pelas colunas Nome e NIF e o processo é feito com a seguite *query*:

SELECT nome AS "Nome", nif AS "nif" FROM empresa ORDER BY nome

Na **Categoria** apenas a designação aparece e tem como *query:*

SELECT designacao AS "Designação" FROM categoria ORDER BY designação

Já em **Manutenções** aparece a lista de todos os processos de manutenção com os valores Tarefa, Procedimento, Periodicidade e Custo. A sua *query* é:

SELECT tarefa AS "Tarefa", procedimento AS "Procedimento", periodicidade AS "Periodicidade", custo AS "Custo" FROM lips ORDER BY tarefa

7.5.10.3. Relatório de Actividades

Um gestor de manutenção tem a necessidade de elaborar um relatório de actividades relativo a um intervalo de tempo, vulgarmente um ano. Para isso, o software tem a capacidade de efectuar uma listagem de todas as manutenções efectuadas num período de tempo. Para isso, clica-se no menu **Avançado** e, de seguida, **Relatório de Actividades** que quando clicado surge uma nova pasta com o nome **Registo de Manutenções**, onde se selecciona a data de início e de fim do relatório pretendido, como se mostra na Fig. 48.

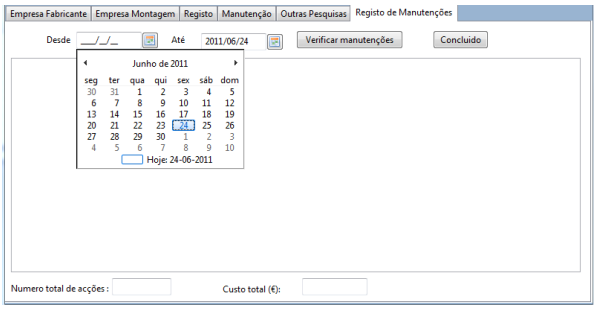
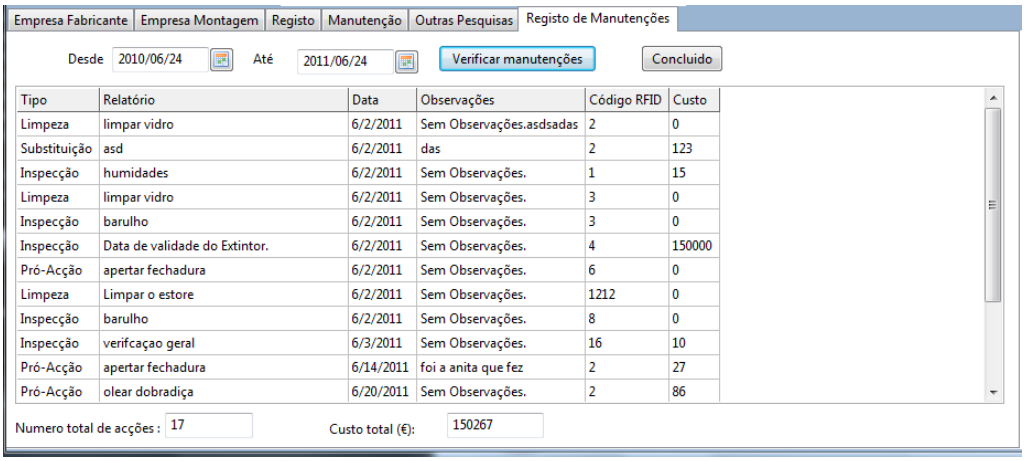


Fig. 48 - Pasta Registo de Manutenções do programa MainTech (selecção da data).

Depois de seleccionadas as datas, o software produz um *output* em forma de tabela com as informações de todas as acções de manutenção nesse intervalo de tempo.



Tipo	Relatório	Data	Observações	Código RFID	Custo
Limpeza	limpar vidro	6/2/2011	Sem Observações.asdsadas	2	0
Substituição	asd	6/2/2011	das	2	123
Inspecção	humidades	6/2/2011	Sem Observações.	1	15
Limpeza	limpar vidro	6/2/2011	Sem Observações.	3	0
Inspecção	barulho	6/2/2011	Sem Observações.	3	0
Inspecção	Data de validade do Extintor.	6/2/2011	Sem Observações.	4	150000
Pró-Acção	apertar fechadura	6/2/2011	Sem Observações.	6	0
Limpeza	Limpar o estore	6/2/2011	Sem Observações.	1212	0
Inspecção	barulho	6/2/2011	Sem Observações.	8	0
Inspecção	verifcaçao geral	6/3/2011	Sem Observações.	16	10
Pró-Acção	apertar fechadura	6/14/2011	foi a anita que fez	2	27
Pró-Acção	olear dobradiça	6/20/2011	Sem Observações.	2	86

Fig. 49 - Pasta Registo de Manutenções resultados do programa MainTech.

Neste *output* tem-se uma tabela onde se encontram as informações Tipo, Relatório, Data, Observações, Código RFID e Custo de todas as manutenções efectuadas no período de tempo seleccionado. Informações como número total de acções e o custo total destas também estão presentes no final da tabela.

A *query* gerada pelo software é dependente das datas seleccionadas, e apenas inclui os registos cujo estado seja Executado, ficando então:

SELECT tipo AS "Tipo",relatorio AS "Relatório", data_reg AS "Data", observ AS "Observações", ref_ele AS "Código RFID" ,custo AS "Custo" FROM registo WHERE data_reg>'"+DateEditDesde.text+"' AND data_reg<'"+DateEditAte.text+"' AND estado="3"

7.5.10.4. Garantias

Outra das capacidades do software é a de verificar quais as garantias que estão para terminar num dado período de tempo, isto é, permite verificar se a garantia de cada elemento irá terminar num determinado período de tempo escolhido pelo utilizador. Para isto clica-se no menu **Avançado** e, de seguida, em **Garantias**. Uma nova pasta com o nome **Garantias** aparece onde se pode seleccionar se se pretende verificar todas as categorias ou uma categoria específica. Selecciona-se o intervalo de tempo desejado e clica-se em **Verificar Garantias**.

O software irá executar um ciclo de verificação se elementos dentro da garantia actualmente, estarão depois do período de tempo seleccionado da seguinte forma:

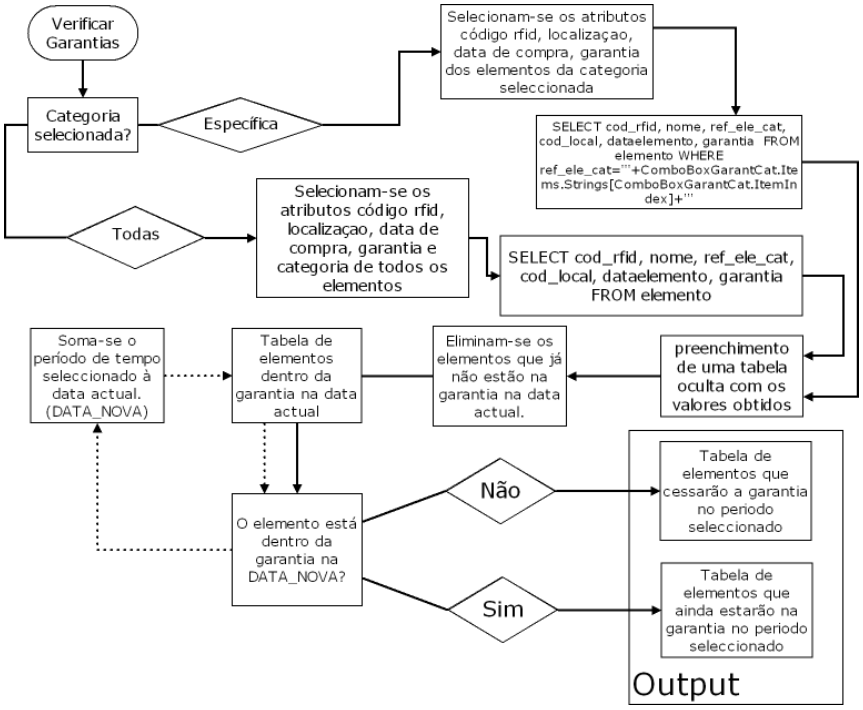


Fig. 50 - Organograma da verificação de garantias.

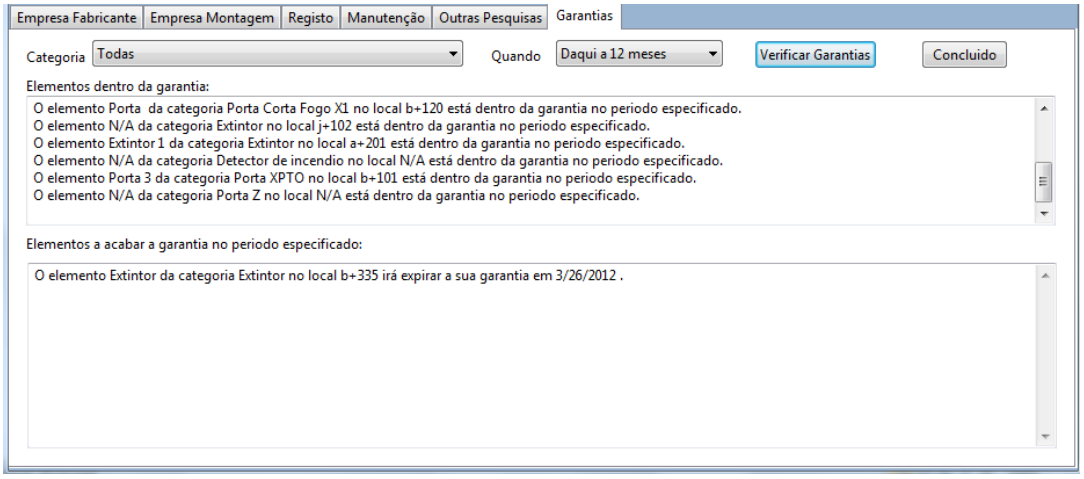


Fig. 51 - Pasta Garantias do programa MainTech, exemplo de output.

O *output* será dado em duas caixas de texto, onde, na primeira, aparece uma lista de todos os elementos com garantia nesse período de tempo e, na segunda, aparece uma lista dos elementos cuja garantia cesse no período especificado.

7.6. CONCLUSÕES

A aplicação MainTech tem capacidades de simplificar processos da manutenção de edifícios de que se salientam:

A obtenção de todas as informações relativas a manuais de manutenção no local do elemento é uma mais-valia para a manutenção pois, deste modo, o operador tem toda a informação necessária à operação que irá executar, não existindo equívocos nem acções executadas de forma errada. Tendo todos os registos associados ao elemento, qualquer alteração deste está disponível de forma imediata para o operador. Na perspectiva do gestor de manutenção, tem-se a vantagem de este ter a informação de todas as acções realizadas de uma forma normalizada. A realização de um relatório de actividades automático por parte do software não só simplifica as obrigações do gestor como também o faz sem possíveis esquecimentos. A verificação de elementos simplifica um processo que, de um modo não informático, seria extremamente demorado e que, com recurso a meios informáticos, permite obter esta informação de um modo rápido e fidedigno.

A única diferença que se prevê existir entre a utilização do leitor utilizado e o leitor seleccionado no capítulo 5.6 é o facto de um leitor com maior distância de leitura efectuar a leitura de vários *tags* de uma só vez. Desta forma, o software necessitaria de um componente adicional em que fossem listados todos os elementos lidos pelo leitor permitindo ao operador escolher entre os elementos lidos.

8
CONCLUSÕES

8.1. CONSIDERAÇÕES FINAIS

No final dos capítulos 5, 6 e 7 encontram-se conclusões detalhadas de cada um, que aqui se sintetizam.

Conclui-se neste trabalho que um sistema de identificação por RFID tem grandes capacidades quer em termos de características quer no que diz respeito à relação custo/benefício. A identificação por RFID pode vir a colmatar uma lacuna na optimização de custos no âmbito da manutenção de edifícios, até agora não preenchida devido à falta de tecnologia. Sem a tecnologia RFID todos os outros sistemas de identificação automáticos, tais como código de barras ou QR Code, seriam obsoletos pois ter-se-ia a necessidade destes identificadores estarem no exterior do elemento. O que possibilitaria que estas identificações fossem facilmente danificadas pois estariam expostos a diversos factores de deterioração, sendo mesmo susceptíveis de ser removidos ou danificados devido a actos de vandalismo.

A tecnologia para a realização de uma base de dados tem sido desenvolvida desde os anos 60. Mas a utilização de uma base de dados sem um meio rápido e eficaz de identificar o elemento pode tornar esta ferramenta num aumento de dificuldade à manutenção ao invés de a facilitar. Conjugando uma base de dados com uma identificação automática à distância aumenta a sua utilidade e facilidade de utilização, podendo deste modo aproveitar todo o potencial de possuir toda a informação normalizada relativa à manutenção de um edifício.

A tecnologia RFID, apesar de estar em desenvolvimento contínuo desde a Segunda Guerra Mundial, tem ainda um grande potencial de desenvolvimento. Cada vez mais, os produtos associados à tecnologia têm preços mais reduzidos, muito devido à adopção desta tecnologia por parte da indústria. Os custos associados à implementação deste sistema irão depender do grau de exigência como verificado no capítulo 5.4. Para edifícios em período de projecto, o custo da adopção da tecnologia é bastante reduzido. A adopção desta tecnologia para edifícios em período de utilização é bastante limitada, pois a apenas se poderiam colocar *tags* em alguns elementos, mas quem, em certos casos poderia ainda seria proveitoso. Dando como exemplo, apesar de não se poder efectuar uma manutenção generalizada do edifício apoiada neste sistema, poder-se-ia apenas associar ao sistema alguns elementos como extintores, AVAC, elevadores, etc.

Existem várias vantagens directamente ligadas ao uso do sistema:

- Informação de todos os elementos fonte de manutenção normalizada.
- Acesso a informação de proximidade, como, por exemplo, o cadastro de acções e manuais de manutenção.

- A abertura de um registo só é possível com a proximidade do operador com o elemento, o que implica a obrigatoriedade do mesmo se deslocar efectivamente ao local impedindo operações de manutenção fraudulentas.
- Substituições falaciosas seriam facilmente detectáveis.
- Controlo da produtividade dos operários.
- Melhor organização para planos de trabalhos.
- Melhor previsão de custos.
- Empresas podem receber informações para melhorar os seus produtos.

Dando um exemplo em termos económcios, toma-se um projecto hipotético e simplificado de uma faculdade, com 200 salas de aulas e 100 salas de apoio. O somatório dos elementos identificados para cada sala será aproximado tendo cada sala identificado por RFID os seguintes elementos: duas portas, quatro paredes, uma cobertura, um soalho, quatro janelas, dois detectores de incêndio, um extintor, dois interruptores, 15 candeeiros e um ar condicionado sendo um total de trinta e três *tags*. Utilizando um preço aproximado de 0,50€ por *tag* cada sala teria um custo de 16.50€ e o edifício todo (300 salas) de 4950€. Associando o custo de um leitor, as infra-estruturas do sistema terão um custo total de aproximadamente 5500€. Este valor é praticamente insignificante num projecto deste tamanho. Refira-se que a aquisição de *tags* e leitores na mesma empresa poderia trazer preços significativamente menores.

Sentiu-se, por parte das empresas contactadas (Telemax, Creativesystems, Alien Technology Corporation, GAO RFID Inc., Shenzhen New Force Communication Technology Co.) uma grande disponibilidade para a concepção de novas soluções para diferentes aplicações, estando portanto interessados em desenvolver novos conceitos para ampliar o mercado desta tecnologia.

Para além da evolução tecnológica de que esta área tem sido e continuará a ser alvo, esse progresso alarga-se também ao desenvolvimento de normas da Organização Internacional de Normalização. Como referido no capítulo 4.7, existem normas para cada uma das aplicações onde actualmente é usada esta tecnologia.

Para comprovar as vantagens que a aplicação real de RFID na manutenção de edifícios pode trazer seria necessário um longo período de tempo. Por esta razão, no caso de serem provadas as vantagens, esta tecnologia só gerará o retorno esperado em relação ao investimento inicial a médio ou longo prazo.

8.2. DESENVOLVIMENTOS FUTUROS

O uso desta tecnologia na área de manutenção de edifícios é ainda uma possibilidade, sendo que, para poder funcionar de um modo que o autor considera perfeito, necessita de uma normalização por parte da Organização Internacional de Normalização.

Esta normalização permitiria que os elementos fossem fabricados com um *tag* embutido. Este *tag* seria de uma frequência de operação normalizada, assim como o formato normalizado para cada elemento possível. Cada elemento continha informações fixas próprias, como o lote do fabrico, data, local onde esteve armazenado, detalhes do transporte etc., assim como continha informações gerais sobre os produtos iguais a ele, como manuais utilização, manuais de manutenção, preço, garantia, referência, etc.

Para estas normalizações seriam necessário efectuar vários estudos para optimizar o sistema utilizado para a identificação de cada elemento:

Em primeiro lugar, seria necessário um estudo sobre as informações relevantes a estarem presentes na memória interna da *tag*, assim como a organização de tal informação. Assim, os softwares executados para esta utilização poderiam introduzir directamente o elemento para a base de dados apenas lendo o seu *tag*.

Outro estudo seria testar que tipo de *tag* teria uma melhor performance num dado elemento. Cada elemento é constituído por diferentes materiais, logo o *tag* necessitaria de diferentes tipos de antenas de modo a que a distância de leitura fosse idêntica entre os vários elementos.

A localização desses *tags* passaria também por uma normalização, sendo necessário efectuar um estudo de modo a descobrir qual a localização com melhor performance para cada tipo de elemento diferente.

Com a normalização deste sistema, seria possível a criação de uma base de dados internacional onde constariam informações de performance e custos de manutenção dos mais variados elementos de empresas diferentes.

Neste trabalho descartou-se o uso de *tags* sensores não só pelo seu preço, como também pela necessidade de estes possuírem uma bateria, facto que diminuiria extremamente o período de vida do *tag*. Estes *tags* teriam como função a monitorização de propriedades fundamentais ao bom funcionamento do elemento. Efectivamente existem *tags* com a capacidade de serem alimentados pela energia da rede e, nos elementos electromecânicos, este facto não seria um entrave. Desta forma, considera-se importante um estudo sobre as vantagens da utilização de *tags* activos com sensores em elementos electromecânicos.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] Calejo, R., *Gestão de edifícios – Modelo de Simulação Técnico-económica.* Dissertação de Doutoramento, Faculdade de Engenharia da Universidade do Porto, 2001.

[2] Mota, P., *Gestão de Habitação Social – Proposta de um manual de serviço das zonas comuns.* Dissertação de Mestrado, Faculdade de Engenharia da Universidade do Porto, 2010.

[3] Tavares, A., *Gestão de edifícios – Informação Comportamental*. Dissertação de Mestrado, Faculdade de Engenharia da Universidade do Porto, 2009.

[4] Torres, J., *Manutenção Técnica de Edifícios – Vãos Exteriores: Portas e Janelas*. Dissertação de Mestrado, Faculdade de Engenharia da Universidade do Porto, 2009.

[5] Wikipedia, "Carta de Atenas". Disponível em http://pt.wikipedia.org/wiki/Carta_de_Atenas Acesso a 28 de Março de 2011.

[6] "Carta de Atenas". disponível em http://idarqfactor.org/Carta de Atenas.pdf Acesso a 28 de Março de 2011.

[7] Cordeiro, C. *Análise Comportamental de Edifícios – Observação de custos em serviço.* Dissertação de Mestrado, Faculdade de Engenharia da Universidade do Porto, 2009.

[8] Calejo, R. *Manutenção de Edifícios – Análise e exploração de um banco de dados sobre um parque habitacional.* Dissertação de Mestrado, Faculdade de Engenharia da Universidade do Porto, 1989.

[9] Wikipédia, "Código de Barras". Disponível em http://pt.wikipedia.org/wiki/C%C3%B3digo_de_barras Acesso a 2 de Abril de 2011.

[10] Seideman, T. *Barcodes Sweep the World*, 1993, http://www.barcoding.com/information/barcode_history.shtml Acesso a 2 de Abril de 2011.

[11] Wikipédia, "Código QR". Disponível em http://pt.wikipedia.org/wiki/C%C3%B3digo_QR Acesso a 1 de Junho de 2011.

[12] Roberti, M. *"The History of RFID Technology"*. Disponível em http://www.rfidjournal.com/article/view/1338 Acesso a 4 de Abril de 2011.

[13] Wikipédia, "*RFID*". Disponível em http://pt.wikipedia.org/wiki/RFID Acesso a 21 de Março de 2011.

[14] Wikipédia, "*Radio-frequency identification*". Disponível em http://en.wikipedia.org/wiki/Radio-frequency_identification Acesso a 4 de Abril de 2011.

[15] Wikipédia, "*Transponder"*. Disponível em http://pt.wikipedia.org/wiki/Transponder Acesso a 28 de Abril de 2011.

[16] Bonsor, K., Keener, C. "*How RFID Works".* 2007. Disponível em http://electronics.howstuffworks.com/gadgets/high-tech-gadgets/rfid.htm Acesso em 5 de Abril de 2011.

[17] Dobkin, D. "*RFID Basics: Antenna Polarization"* 23 de Outubro de 2007. Disponível em http://www.eetimes.com/design/microwave-rf-design/4018934/RFID-Basics-Antenna-Polarization Acesso a 8 de Abril de 2011.

[18] http://www.apfm.pt/ Acesso a 28 de Março de 2011.

[19] http://rfidtek.blogspot.com/2009/09/protocolos-e-normas-epc.html Acesso a 1 de Junho de 2011.

[20] Turner, C. "*EPC and ISO 18000-6*" Disponível em http://www.rfidjournal.com/article/view/325 Acesso a 2 de Junho de 2011.

[21] Wikipédia, "*Chave Primária*". Disponível em http://pt.wikipedia.org/wiki/Chave_prim%C3%A1ria Acesso a 2 de Junho de 2011.

[22] Wikipédia, "*Banco de Dados*". Disponível em http://pt.wikipedia.org/wiki/Banco_de_dados Acesso a 2 de Junho de 2011.

ANEXOS

CONSIDERAÇÕES

Dado o código fonte da aplicação criada (MainTech) ser de grande extensão, não se considerou incluir uma cópia escrita do código na totalidade, referenciando partes de código quando necessário. Tomou-se esta decisão não só pelo grande número de páginas ocupado mas também pelo facto de que a necessidade do mesmo seria mais útil em formato digital. Por isso, este encontra-se no CD incluído.

O CD incluído contém também o código de criação da Base de Dados, um formato digital do texto, a aplicação MainTech compilada e um manual de instalação no sistema.

ANEXO 1

FICHA TÉCNICA
MOTOROLA MC3090-Z

SPECIFICATION SHEET

MC3090-Z

Business-class handheld RFID reader

FEATURES

Lightweight, ergonomic pistol grip design
Built for all day comfort; reduces user fatigue in read- and scan-intensive applications

Motorola's patented omnidirectional MAX *RFID Antenna*
Orientation insensitive design eliminates the need for precise reader and tag alignment for faster and more accurate reads

Color touchscreen
Easy and intuitive interface; reduces training time and costs

Pinpoint locationing technology
A unique combination of intuitive audible and visual cues easily guides workers to a specific item: as the device moves closer to the desired item, a beep tone increases in frequency and volume, while a sliding graphical bar increases in size

New class of RFID reader brings the benefits of RFID from industrial environments to the carpeted space

The MC3090-Z represents another RFID first from Motorola — the first business-class handheld RFID reader designed specifically to extend the benefits of RFID beyond industrial spaces and into customer facing and business environments. The MC3090-Z starts with the signature rugged design and high performance for which Motorola's industrial handheld RFID products are known, and adds the ergonomics required for all day comfort and ease of use. At just half the weight of its industrial counterparts, the MC3090-Z is the lightest UHF RFID rugged handheld reader on the market, offering a well-balanced design and a convenient gun-style grip for read intensive applications. Add a groundbreaking new antenna design that drives RFID performance to new heights and the result is a highly versatile device that is at home in customer facing environments — from retail stores and healthcare facilities to the business office.

Motorola MAX *RFID Antenna* — delivering maximum RFID performance

Until now, enterprises have been forced to choose between two types of antennas: linear polarization for a longer read range or circular polarization for wider coverage. The MC3090-Z antenna combines the advantages of these two technologies into a patented omnidirectional antenna that offers the best of both worlds — a superior read range and a superior coverage area. The orientation insensitive antenna delivers extraordinary reliability and eliminates the need to precisely align the reader with the tag, adding to user comfort and productivity. The result is the ability to accurately and rapidly capture RFID tags on even the most challenging items — from a pile of clothing in a retail store or a box of files in the office to data tapes in the data center.

Motorola's signature rugged design and dual data capture functionality deliver superior value

The MC3090-Z offers the best of form and function with Motorola MAX *Rugged* and Motorola MAX *Data Capture*. While the MC3090-Z is designed for the carpeted space, Motorola MAX *Rugged* provides Motorola's flagship rugged specifications, ensuring dependable operation and a maximum lifecycle in demanding business environments. The reader passes one of the industry's most stringent impact tests, able to survive a 4 ft./1.2 m drop to concrete across the entire operating temperature range. In Motorola's unique endurance test, the MC3090-Z continued to perform reliably after 1,000 1.64 ft./0.5m drops in Motorola's tumble drum. And IP54 sealing enables the device to withstand dusty environments, spills and the routine wipedowns required in healthcare and other challenging environments.

Motorola MAX *Data Capture* offers two best-in-class advanced data capture technologies in one device — the ability to read RFID tags and scan bar codes. The number of devices you need to

Easy-to-use Application Programming Interfaces (APIs)
Enables rapid and cost-effective application development

Wi-Fi 802.11a/b/g wireless connectivity
Connects to virtually any wireless LAN for easy integration into any wireless enterprise environment; enables real-time wireless data capture anywhere in your environment

Motorola MAX *Rugged*: meets and exceeds MIL-STD 810F standards for drop, tumble and sealing
Built for all day enterprise use; provides dependable operation despite the inevitable drops and spills; provides extended lifecycle; withstands wipe downs

Motorola MAX *Data Capture*
Offers exceptional RFID and bar code scanning functionality in a single device; eliminates the need to purchase two devices, reducing capital and operational costs

320 x 320 display with backlight
30% higher resolution than ¼ VGA for easier viewing in virtually any lighting condition

purchase and manage is reduced — as well as the associated capital and operational costs. In addition to exceptional RFID performance, the device also enables first time capture of even poor quality 1D bar codes, further protecting user productivity and data accuracy.

Easy-to-use robust locationing

Until now, RFID handheld readers identified the general proximity of an item — for example, a shelf. The MC3090-Z is the only RFID reader to offer pinpoint locationing technology. A unique combination of intuitive audible and visual cues quickly and easily guides workers to the desired item. As the device moves closer to a specific item, a beep tone increases in frequency and volume, while a sliding graphical bar increases in size.

The Motorola end-to-end advantage

When you choose the Motorola MC3090-Z, you enjoy the advantages of a world-class partner channel, world-class management solutions and world-class services. Our award-winning partner channel offers a best-in-class broad set of ready-to-go and custom applications, minimizing deployment time and cost. Compatibility with Motorola's Mobility Services Platform (MSP) and the Motorola Mobility Suite offers extraordinary centralized control over all your Motorola devices — the ability to remotely stage, provision, monitor, troubleshoot, secure data on the devices and much more drives device management costs to a new low. And since your employees will count on the MC3090-Z every day, Motorola's Service from the Start with Comprehensive Coverage will help keep your devices up and running at peak performance. This exceptional service is truly comprehensive, providing technical software support and software downloads as well as true end-to-end and inside-outside protection for your device. Normal wear and tear, internal and external components damaged through accidental breakage and select accessories that ship together with the MC3090-Z are all covered — at no additional charge.

For more information on how you can bring the benefits of RFID into new areas of your business, please visit us on the web at www.motorola.com/MC3090Z or access our global contact directory at www.motorola.com/enterprisemobility/contactus

MC3090-Z Applications

As the first business-class handheld RFID reader, the MC3090-Z enables the extension of RFID beyond industrial spaces and into customer facing and business environments. Industries and applications include:

Industry	Application
Retail Sales Floor; POS; Backroom	• Inventory/Cycle Counting • Automatic Replenishment • Receiving/Shrink Control • Item Finding
Enterprise Business office/carpeted space	• Asset/IT Management • File and Document Tracking • Item Finding
Healthcare Hospitals; clinics	• Asset Management (for high value critical assets) • Patient Tracking • Item Finding
Manufacturing/Warehousing (environmentally controlled) Production line; warehouse aisles and loading docks	• Work In Process (WIP) • Receiving/Shipping

1

ANEXO 2

FICHA TÉCNICA
SYSTEM LIMITED CS 101

CSL CONVERGENCE SYSTEMS LIMITED

CS101 Handheld RFID Reader

Product Profile:

CS101 is a UHF EPC C1G2 handheld RFID reader with industry leading read range and read rate performance to enable ultra fast inventory of cases on pallet, warehouse shelves, moving vehicles, and other tagged items in business processes where fixed reader class performance is required in a portable device.

Features:

- Excellent read range: 7 meters open space, 7 - 11 meters indoor confined space
- High read rate: typical 150 tags per second for selected Gen2 profiles (peak at 400 tags/s)
- Country specific versions in terms of frequency bands
- Choice of vertically linear polarized or horizontally linear polarized antenna
- Ruggedized design: Drop Test 1.5 meters on 6 surfaces, IP65

Specifications:

Physical Characteristics:	Length = 197 mm, Width = 122 mm; Height = 223 mm; Weight = 1Kg
Read Range:	Up to 7 meters in outdoor or open area (using AD431 tag) 7 - 11 meters in indoor confined space (using AD431 tag)
Read Rate:	Typical 150 tags per second for selected Gen2 profiles (peak at 400 tags/s)
Protocol:	ISO18000-6C, EPC UHF Class 1 Gen 2
Tag Information:	All banks, any sizes (e.g. 240 bits bank1, 512 bits bank3), Per 'Tag Read' RSSI value
Frequency Range:	All frequency bands in the world available
Barcode Scanner:	1D and 2D: all standards available
Antenna Polarization:	Choice of vertically linear polarized or horizontally linear polarized
Environment:	Operating Temp: -20°C to 50°C (-4°F to 122°F) Storage Temp: -40°C to 75°C (-40°F to 167°F) Humidity: 5% to 95% Non-condensing
Ruggedized Designs:	Drop test 1.5 meters on 6 sides; IP65
User Interface - Input:	QWERTY keyboard, Touch Screen, Trigger Button
User Interface - Output:	3.5" Transmissive LCD, 320 x 240 pixels, LED x 2, Buzzer
Connectivity:	Wi Fi (802.11 b/g), USB, RS232, GPRS (with CS501 module)
External Memory:	SD Card, Standard SD and SDHC
Processor & Memory:	Samsung, 400 MHz, 128 MByte Flash, 64 MByte RAM
Battery Life:	1.5 hours continuous read, 20 hours standby, easily field replaceable battery
Source Code Samples:	Read tag, write tag, tag inventory, tag ranging, tag search, tag authentication, tag association, tag commissioning, barcode scanning, database manipulation, wireless network transfer, server side driver, etc.
Order Code:	CS101-V/H-N N=1: 865-868 MHz (CE for Europe) & 865-867 MHz (for India), N=2: 902-928 MHz (FCC for USA), N=3: 952-954 MHz (Telec for Japan), N=4: 919-928 MHz (SRRC for China, HK, Malaysia, Singapore, NCC for Taiwan, etc.); (V for vertically linear polarized, H for horizontally linear polarized)

CHUNG NAM BUILDING,
20th Floor,
No. 1 Lockhart Road,
Wanchai
Hong Kong
TEL: (852) 25293008
FAX: (852) 26832018
WEBSITE:
www.convergence.com.hk

ANEXO 3

FICHA TÉCNICA
AVERY DENNINSON AD-223

AD-223

AVERY DENNISON

Print

Antenna Size:
3.741 x 0.321 in
[95 x 8.15mm]

Features:

- Best read range performance in the AD-22x family
- All-purpose transponder tuned for global supply chain application use
- Write range approaches read range
- Tight pitched high volume rolls
- Available in high yield/capacity rolls which integrate seamlessly into high-volume converting processes

Market Applications:

Electrical Characteristics	
Operating Frequency	Global (860 - 960MHz)
Integrated Circuit [IC] – Manufacturer & IC Name	Impinj Monza 3
RF Communications Protocol	ISO/IEC 18000-6C & EPCglobal Class 1 Gen 2
Memory	96 bits EPC
Environmental	
Operating Temperature	-40°F to 185°F [-40°C to 85°C]
RoHS	Designed to comply with EU Directive 2002/95/EC
Inlay Roll Format / Finishing	
Un-wind direction	Inlay-side out
Core size with adaptor insert	6 in [152 mm]
Maximum roll outer diameter [not to exceed]	12 in [305 mm]

Part Details for AD-223 (Total 3 Parts)

Part Number	**600192**
Chip	Impinj Monza 3
Antenna dimensions, [CDxMD]	3.741 in [95.03mm] x .321 [8.15mm]
Die-cut dimensions	N/A
Inlay-to-liner adhesive	N/A
Liner material	N/A
Total thickness over chip [typical]	11.5 mils [292 microns]
Standard pitch	.625 in [15.88mm]
Standard web widths	4.75 in [120.65mm]
Inlay substrate material	PET
Quality assurance	100% read tested with out-of-tolerance inlay marked
Average number of units per roll	20,000
Rev	03