

**Universidade do Minho**

Nuno Manuel Rodrigues Vieira

**Monitorização e prevenção de  
falhas em sistemas baseados em Agentes**

Protótipo da Aplicação

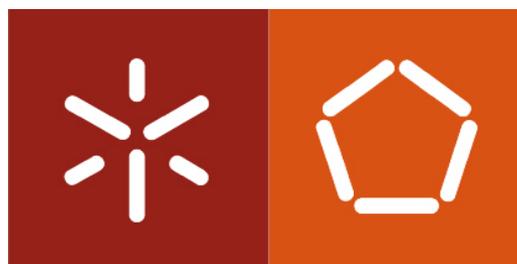
**Tese de Mestrado**

Mestrado Integrado em Engenharia Biomédica

Trabalho efetuado sob a orientação de:

**Professor Doutor José Machado**

2 de Novembro de 2012



Escola de Engenharia  
**Universidade do Minho**

Monitorização e Prevenção de Falhas em Sistemas baseados em Agentes

Vieira, Nuno R V. (2012). Monitorização e Prevenção de Falhas em Sistemas baseados em Agentes. Dissertação apresentada à Universidade do Minho como requisito à obtenção do grau de Mestre em Engenharia Biomédica - Ramo de Informática Médica, realizada sob a orientação do Professor Doutor José Machado, Professor Associado com Agregação do Departamento de Informática da Universidade do Minho.



# Agradecimentos

Ao fim de uma etapa importantíssima da minha vida, como é o concluir de uma dissertação, seria necessário o agradecimento a uma lista infindável de pessoas que mereciam ser consideradas e lembradas nestes agradecimentos. No entanto, não posso deixar de agradecer à minha família, em especial aos meus Pais, a quem dedico esta dissertação. Por todo o apoio dado, principalmente ao longo de todo o percurso acadêmico, pois obviamente sem eles, nada disto seria possível.

Ao Professor Doutor José Machado, pela orientação da dissertação e, sobretudo, pelo seu precioso tempo disponibilizado. Agradeço todos os ensinamentos e experiência transmitida.

Ao colega e amigo, Luís André, pela ajuda no trabalho desenvolvido, pelos conhecimentos transmitidos, e pelo apoio ao longo de toda a dissertação.

À minha irmã, Isabel, pelo auxílio e motivação dada e também pela paciência disponibilizada em determinados momentos, os quais considero que deveriam e poderiam ter sido mais.

À Cátia, pela ajuda e apoio em momentos de maior dificuldade que este trabalho acabou por trazer, sendo assim uma motivação para continuar.

Por fim, ao colega e sempre amigo Alison Gomes pela preciosa ajuda na tradução do Resumo para a língua Inglesa.

A todos, o meu muito Obrigado.



# Resumo

Os avanços tecnológicos verificados nos dias de hoje assim como a quantidade de informação e comunicação que lhes estão associados, atribuem um papel de grande importância aos sistemas de monitorização. É no seio desta evolução, que a competição existente entre os vários setores de mercado não estão acessíveis a erros e falhas, principalmente ao nível dos equipamentos tecnológicos. Assim, neste contexto de intolerância, assiste-se a uma proliferação e a uma utilização cada vez maior de sistemas de monitorização e prevenção.

Mais importante que monitorizar é prevenir. Ter a capacidade de evitar uma falha, e permitir a resolução de algum problema atempadamente é uma mais valia para o desempenho, atribuindo fiabilidade e qualidade ao serviço. A gestão e a verificação de equipamentos, bem como de processos associados aos mesmos permitem um maior controlo e domínio dum sistema.

Assim esta dissertação tem como objetivo principal a implementação dum sistema para monitorizar a atividade de um ou mais sistemas multi-agente, com capacidade para intervir e avisar o administrador do sistema quando ocorre um problema.

O sistema construído, que assenta numa estrutura formada por três unidades distintas, unidade de análise, unidade de processamento e unidade de interface com o utilizador, permitem a implementação de processos para troca e integração de informação, exercendo assim uma comunicação fundamental entre sistemas e utilizadores. É um sistema direcionado principalmente à gestão e monitorização do desempenho de diferentes equipamentos assim como dos processos em execução nos mesmos. Este trabalho foi desenvolvido

em colaboração com um Hospital no Norte do País e o ambiente escolhido para a implementação da plataforma foi unicamente laboratorial.

A concretização deste projeto esteve dividida em quatro fases: Início, Pesquisas, Construção da Aplicação e Escrita da Dissertação. Na primeira fase, o Início, foi feito o levantamento dos principais requisitos para o sistema, definição dos objetivos, e elaboração de um plano de trabalho. Na segunda fase, Pesquisas, procedeu-se ao levantamento de informação sobre os conceitos teóricos relacionados com o tema em causa, como artigos científicos que dão suporte ao tema, entre outros trabalhos já publicados na mesma área. No final desta etapa já se encontravam definidas as ideias base da aplicação a construir de acordo com as necessidades. Na terceira fase deste projeto, a Construção englobou a modelação e implementação da plataforma de monitorização, de acordo com as especificações definidas anteriormente. A quarta etapa englobou a escrita da dissertação, o que incluiu um enquadramento dos conceitos teóricos em função da aplicação desenvolvida.

## **Palavras-Chave**

Monitorização; Prevenção; Falhas; Multi-Agente

# Abstract

The technological advancements that are present today as well as the quantity of information and communication with them associated, attributes a major role to monitorization systems. It is in the heart of this evolution that competition exists between the various market sectors and so not giving any leeway to weaknesses or mistakes, above all by the technological equipments. And so, in this context of intolerance, a proliferation and utilization of system monitorization and prevention is all the more important.

Prevention is more important than monitorization. Having the capacity to prevent a failure and timely access it with a resolution is of major worth to the quality of the service. The management and verification of these equipments, as well as the processes with them associated allows for a greater regulation and dominion of a system.

Having said this, this dissertation has for its main objective the implementation of a system to monitorize the activity of one or more multi-agent systems with the capacity to intervene and warn the system administrator of when a problem occurs.

The system that was developed was so in a structure with three distinct stages, the stages are: Data Analysis, Data Processing, and User Interface. Together, they allow for an implementation of processes of exchange and integration of information, playing a major role in the communication between users and the system platform. It is a system primarily aimed at the management and monitorization of the performance of equipments as well as of the execution processes. This project was developed in collaboration with an Hospital located on North of Portugal and the chosen conditions for the

implementation of this platform were solely achieved in the laboratory.

The completion of this project was divided into four phases: Beginning, Investigation, Application Development, and the composition of the Dissertation. In the first phase, Beginning, the fundamental system requisites were collected, objectives were defined, and a work plan were elaborated. Then, in the second phase, Investigation, a collection of information regarding theoretical concepts and a vast and varied ammount of information related to the subject was also collected. In addition, information regarding cientific articles in relation to the subject were also collected, as well as information on other projects that were done in relation to the subject and also support it. In the final stages of this phase an application prototype had already been defined also congruent with the necessities that had been discovered within the previous phases. In the third phase of this project, the Development is concerning the elaboration and implementation of the monitorization platform itself, corresponding to the specifications defined previously. The fourth and final phase involved the composition of the dissertation itself, which includes the framework of theoretical concepts directly related to the explanation of the developed application.

### **Key Words**

Monitoring; Preventing; Fault management; Multi-agent

# Conteúdo

|   |            |
|---|------------|
| <b>Lista de Figuras</b>                                       | <b>xiv</b> |
| <b>Lista de Tabelas</b>                                       | <b>xvi</b> |
| <b>1 Introdução</b>   | <b>3</b>   |
| 1.1 Enquadramento . . . . .                                   | 3          |
| 1.2 Estrutura da Dissertação . . . . .                        | 10         |
| <b>2 Trabalho Relacionado</b>                                 | <b>11</b>  |
| 2.1 Sistemas de Informação . . . . .                          | 11         |
| 2.2 Interoperabilidade . . . . .                              | 15         |
| 2.3 Descoberta de conhecimento em Bases De Dados . . . . .    | 23         |
| 2.4 <i>Data Mining</i> . . . . .                              | 25         |
| <b>3 Visão do Produto</b>                                     | <b>29</b>  |
| 3.1 Visão Global . . . . .                                    | 29         |
| 3.2 Estrutura e Constituintes . . . . .                       | 31         |
| 3.2.1 Unidade de Análise . . . . .                            | 32         |
| 3.2.2 Unidade de Processamento . . . . .                      | 34         |
| 3.2.3 Aplicação <i>Web</i> /Interface de Utilizador . . . . . | 36         |
| 3.3 Características do Produto . . . . .                      | 37         |
| 3.4 Síntese . . . . .   | 38         |
| <b>4 Modelação do Produto</b>                                 | <b>39</b>  |
| 4.1 Metodologias . . . . .                                    | 39         |
| 4.2 Utilizadores . . . . .                                    | 42         |

|          |   |           |
|----------|---|-----------|
| 4.3      | Casos de Uso . . . . .                                | 42        |
| 4.3.1    | Caso de Uso Efetuar Registo . . . . .                 | 44        |
| 4.3.2    | Caso de Uso Login . . . . .                           | 45        |
| 4.3.3    | Caso de Uso Logout . . . . .                          | 46        |
| 4.3.4    | Diagrama Caso de Uso Ver Máquina . . . . .            | 47        |
| 4.3.5    | Diagrama Caso de Uso Notificações . . . . .           | 48        |
| 4.4      | Requisitos . . . . .                                  | 50        |
| 4.4.1    | Requisitos Funcionais . . . . .                       | 50        |
| 4.4.2    | Requisitos Não Funcionais . . . . .                   | 56        |
| 4.5      | Síntese . . . . .                                     | 57        |
| <b>5</b> | <b>Desenvolvimento do Produto</b>                     | <b>59</b> |
| 5.1      | Linguagens Utilizadas . . . . .                       | 59        |
| 5.2      | Frameworks e APIs Utilizadas . . . . .                | 61        |
| 5.3      | Componentes Principais . . . . .                      | 66        |
| 5.3.1    | Master Page . . . . .                                 | 66        |
| 5.3.2    | Autenticação . . . . .                                | 67        |
| 5.3.3    | Monitorização em tempo real . . . . .                 | 68        |
| 5.3.4    | Histórico dos dados monitorizados . . . . .           | 69        |
| 5.3.5    | Agendar Notificação . . . . .                         | 71        |
| 5.3.6    | Envio de Email . . . . .                              | 73        |
| 5.3.7    | Lista de Notificações enviadas . . . . .              | 74        |
| 5.3.8    | Gerador de Relatórios . . . . .                       | 74        |
| 5.4      | Interfaces de Utilizador . . . . .                    | 75        |
| 5.4.1    | Interface Administrador . . . . .                     | 75        |
| 5.5      | Implementação da Base de Dados . . . . .              | 76        |
| <b>6</b> | <b>Análise do Produto - Simulação e Caso de Teste</b> | <b>81</b> |
| 6.1      | Introdução . . . . .                                  | 81        |
| 6.2      | Cenário . . . . .                                     | 82        |
| 6.3      | Procedimento Efetuado . . . . .                       | 83        |
| 6.4      | Conclusão da Simulação . . . . .                      | 84        |

|                                       |           |
|---------------------------------------|-----------|
| <i>CONTEÚDO</i>                       | xi        |
| <b>7 Conclusões e Trabalho Futuro</b> | <b>87</b> |
| 7.1 Conclusões . . . . .              | 87        |
| 7.2 Trabalho Futuro . . . . .         | 89        |
| <b>A Manual de Utilizador</b>         | <b>97</b> |



# Lista de Figuras

|     |  |    |
|-----|--|----|
| 1.1 | Arquitetura de um sistema com interação, comunicação e gestão através da Internet[1] . . . . .               | 8  |
| 1.2 | Esquema da AIDA - Agência para a Integração, Difusão e Arquivo de Informação Médica [2] . . . . .            | 9  |
| 1.3 | Módulos existentes na AIDA - Agência para a Integração, Difusão e Arquivo de Informação Médica [2] . . . . . | 9  |
| 2.1 | Esquema de uma interface de agente comum . . . . .   | 17 |
| 2.2 | Processo DCBD segundo Fayyad et al. . . . .  | 25 |
| 2.3 | Esquema sucinto do termo <i>Data Mining</i> . . . . .  | 28 |
| 3.1 | Esquema representativo do funcionamento global da plataforma de monitorização . . . . .                      | 30 |
| 3.2 | Estrutura da aplicação com os seus diferentes constituintes . . . . .  | 32 |
| 3.3 | Esquema representativo da unidade de análise e seus diferentes constituintes . . . . .                       | 34 |
| 3.4 | Esquema representativo da unidade de processamento e as suas funções. . . . .                                | 36 |
| 4.1 | Diagrama Casos de Uso da aplicação de Monitorização . . . . .  | 44 |
| 4.2 | Diagrama Case de Uso Ver Máquina . . . . .   | 48 |
| 4.3 | Caso de Uso Notificações . . . . .   | 49 |
| 5.1 | Ambiente do Visual Studio durante o desenvolvimento da aplicação em ASP.NET[3] . . . . .                     | 62 |

|      |  |    |
|------|--|----|
| 5.2  | Arquitetura WMI - Relação entre a infra-estrutura WMI, WMI <i>providers</i> e objetos monitorizados[4] . . . . . | 65 |
| 5.3  | Esquema da arquitetura ADO.NET[5] . . . . .  | 66 |
| 5.4  | Esquema de uma Master Page e seus constituintes[6] . . . . .   | 67 |
| 5.5  | Gráficos da monitorização em tempo real, para Processador e RAM . . . . .  | 68 |
| 5.6  | Percentagens livre e ocupado em tempo real, para Processador e RAM e Disco . . . . .                             | 69 |
| 5.7  | Gráficos de dados históricos, para Processador e RAM . . . . .   | 70 |
| 5.8  | Gráficos de dados históricos, para Processador e RAM . . . . .   | 70 |
| 5.9  | Interface para a opção Agendar Notificação . . . . .   | 72 |
| 5.10 | Exemplo de <i>drop down list</i> constituída pela informação existente na base de dados . . . . .                | 72 |
| 5.11 | Campos a utilizar para eliminar uma notificação agendada . . . . .   | 73 |
| 5.12 | Lista das notificações enviadas para os utilizadores . . . . .   | 74 |
| 5.13 | Página principal para Administrador . . . . .  | 76 |
| 5.14 | Esquema relacional da base de dados desenvolvida para suporte a plataforma . . . . .                             | 79 |

# Lista de Tabelas

|      |  |    |
|------|--|----|
| 2.1  | Resumo da segurança associada a um Sistema de Informação, com as ameaças e possíveis soluções [7]. . . . . | 15 |
| 2.2  | Níveis de interoperabilidade segundo Mykkanen and Tuomai- nen (2008)[8] . . . . .                          | 22 |
| 4.1  | Fluxo de eventos relativos ao Caso de Uso Efetuar Registo . . .  | 45 |
| 4.2  | Fluxo de eventos relativos ao Caso de Uso Login . . . . .  | 46 |
| 4.3  | Fluxo de eventos relativos ao Caso de Uso Logout . . . . .   | 46 |
| 4.4  | Requisitos Funcionais: Identificador 1 . . . . .   | 50 |
| 4.5  | Requisitos Funcionais: Identificador 2 . . . . .   | 50 |
| 4.6  | Requisitos Funcionais: Identificador 3 . . . . .   | 51 |
| 4.7  | Requisitos Funcionais: Identificador 4 . . . . .   | 51 |
| 4.8  | Requisitos Funcionais: Identificador 5 . . . . .   | 51 |
| 4.9  | Requisitos Funcionais: Identificador 6 . . . . .   | 52 |
| 4.10 | Requisitos Funcionais: Identificador 7 . . . . .   | 52 |
| 4.11 | Requisitos Funcionais: Identificador 8 . . . . .   | 52 |
| 4.12 | Requisitos Funcionais: Identificador 9 . . . . .   | 53 |
| 4.13 | Requisitos Funcionais: Identificador 10 . . . . .  | 53 |
| 4.14 | Requisitos Funcionais: Identificador 11 . . . . .  | 53 |
| 4.15 | Requisitos Funcionais: Identificador 12 . . . . .  | 54 |
| 4.16 | Requisitos Funcionais: Identificador 13 . . . . .  | 54 |
| 4.17 | Requisitos Funcionais: Identificador 14 . . . . .  | 54 |
| 4.18 | Requisitos Funcionais: Identificador 15 . . . . .  | 55 |
| 4.19 | Requisitos Funcionais: Identificador 16 . . . . .  | 55 |
| 4.20 | Requisitos Funcionais: Identificador 17 . . . . .  | 55 |

|      |   |           |    |
|------|---|-----------|----|
| 4.21 | Requisitos Não Funcionais: Identificador 18 | . . . . . | 56 |
| 4.22 | Requisitos Não Funcionais: Identificador 19 | . . . . . | 56 |
| 4.23 | Requisitos Não Funcionais: Identificador 20 | . . . . . | 56 |

# Lista de Abreviaturas

**IT** Information Technology;

**AIDA** Agência para a Integração, Difusão e Arquivo de Informação Médica;

**DM** Data Mining;

**EHR** Electronic Health Record;

**LIS** Laboratory Information System;

**PHR** Personal Health Records;

**IEEE** Institute of Electrical and Electronics Engineers;

**DCBD** Descoberta de Conhecimento em Bases de Dados;

**ASP** Active Server Pages;

**HTML** HyperText Markup Language;

**PHP** Hypertext Processor;

**SQL** Structured Query Language;

**CIM** Common Information Model;

**API** Application Programming Interface;

**IP** Internet Protocol;

**SNS** Serviço Nacional de Saúde;

**SGBD** Sistema de Gestor de Bases de Dados;

**ADO** ActiveX Data Objects;

**OIS** Operations Information Systems;

**MIS** Management Information Systems;

**CSS** Cascading Style Sheets;

**XML** Extensible Markup Language

**AJAX** Asynchronous JavaScript and XML

**PDF** Portable Document Format



# Capítulo 1

## Introdução

Neste capítulo será feita uma abordagem atual, teórica e social ao tema deste projeto, onde serão apresentados os principais problemas existentes na área, assim como as dificuldades com que as instituições de saúde e a sociedade se deparam. Por conseguinte, será apresentada a motivação para a realização deste projeto, onde a sua introdução iria trazer melhorias significativas para os referidos problemas. É apresentada também a estrutura da dissertação bem como um enquadramento para a elaboração da mesma. A realização desta tese, visa a obtenção do grau de Mestre no Curso de Engenharia Biomédica, no ramo de Informática Médica, pela Universidade do Minho.

### 1.1 Enquadramento

Os avanços tecnológicos têm, hoje em dia, um grande impacto na sociedade, em vários temas e diferentes áreas de trabalho. A área da informação não é exceção a esta evolução e tem-se tornado, cada vez mais, num argumento principal para as empresas e organizações tirarem vantagem competitiva em investimentos futuros.

O facto é que as instituições, para se tornarem competitivas e sobreviverem no mercado, têm investido cada vez mais em tecnologia de informação

como a única forma de tornar seguro o processo de decisão. A sociedade atual e o ambiente envolvente são dominados pela informação que corre por todo o mundo, pelo que todos os sistemas utilizam e vivem da informação. Assim associado a este quadro justifica-se a crescente importância e afirmação dos sistemas informáticos.

Quando se lida com grande quantidade de informação, como é o caso dos sistemas informáticos, onde se efetuam registos e trocas de milhares de milhões de dados surge o problema do controlo, gestão e monitorização desses mesmos dados, o que leva à origem de erros e falhas.

Nas unidades de Prestação de Cuidados de Saúde, as fontes de informação não fogem à regra do que se verifica nos dias de hoje, sendo complexas e volumosas, usando diferentes sistemas operativos, diferentes linguagens, oferecendo um carácter heterogéneo ao sistema. Os hospitais são ambientes de trabalho complexos, onde se verifica a distribuição de pessoas, de sistemas e de informação, exigindo uma comunicação considerável entre os profissionais e os cenários. Cenários esses que são constituídos por infraestruturas de informação distribuídos no local [9]. As comunicações são, na maior parte dos casos, restringidas por infraestruturas ultrapassadas e muitos dos novos projetos deparam-se com dificuldades financeiras e diversas burocracias [10].

Este é um problema cada vez mais comum nos dias de hoje. Do ponto de vista de gestão comercial, hospitais e clínicas encontram-se numa posição exigente e, de certo modo, ingrata. Pacientes exigem sempre o melhor serviço, acompanhado dos tratamentos/atendimentos mais recentes e com a melhor tecnologia, recusando, muitas vezes, pagar o devido valor pelas exigências feitas [11].

A sociedade em geral, comunicação social e os próprios pacientes vivem, de certo modo, intolerantes ao erro e, à mínima falha, logo surgem as notícias negativas. A par disto, a indústria médico-hospitalar é controlada por uma série de leis e restrições regulamentares com um grau de exigência elevado. No sentido de contrariar esta realidade, pretende-se que este trabalho venha a solucionar ou, pelo menos, ajudar na melhoria das condições de atendimento ao paciente, pois evitando erros e falhas nos sistemas, haverá uma melhor qualidade de atendimento.

A título de exemplo, a qualidade dos cuidados de saúde é uma questão muito importante nos estados EUA, como referenciado em [12]. Dezenas de milhares de norte-americanos morrem a cada ano, e muitos mais sofrem consequências não fatais devido a erros no sistema de saúde [13]. Várias abordagens têm sido direcionadas para resolver este problema.

Um sem número de restrições técnicas e financeiras impedem a homogeneidade entre sistemas do foro clínico, médico e administrativo, sem descorar os problemas inerentes às necessidades funcionais deste tipo de sistemas [10]. Manter os registos médicos e lidar com os dados hospitalares é assim um campo exigente, pelo que requer exatidão, facilidade de acesso, rapidez e eficiência. Estes parâmetros podem de facto ser atingido com uma aplicação do género da que este trabalho se propõe a desenvolver.

Quando se fala em Tecnologias de Informação, e por sua vez em informação médica, associada a um contexto hospitalar, surgem sempre diferentes pontos de vista e discussões éticas acerca deste assunto. Inserido no contexto teórico deste trabalho, a monitorização de máquinas hospitalares implica lidar com as suas informações e conteúdos, e é então inevitável o aparecimento de questões relacionadas sobre as restrições existentes nos dados de um hospital.

As questões por norma levantadas prendem-se sobretudo pelo fato de tornar a informação pessoal médica disponível, onde vários tipos de utilizadores poderão ter acesso em qualquer lado e em qualquer altura. Ao contrário dos EUA, que são possuidores de uma plataforma, onde os utilizadores registados tem acesso a informação médica e registos clínicos, a Europa, e Portugal em concreto, são mais restritivos sobre o acesso aberto a dados médicos e clínicos [14].

Assim, o esforço atualmente desenvolvido na área das TI caminha no sentido de uma mudança significativa nas organizações de saúde e, por sua vez, nos registos hospitalares. Deste modo, pretende-se que, no futuro, exista um intercâmbio eficaz de informações de saúde entre as diferentes organizações, com o objetivo de tornar o acesso aberto e global à informação médica. [14] É também associado a este contexto que surge este trabalho que defende que a saída de informação interna de um hospital de forma segura, porém acessível

aos utilizadores, permitiria um conhecimento precoce do erro. Por conseguinte, este seria resolvido mais rapidamente ou até mesmo evitado, uma vez que o profissional de saúde poderia gerir informações em qualquer lugar, não estando restringido ao seu trabalho em ambiente hospitalar.

A partilha de dados e informações clínicas, por meio eletrónico, pode desempenhar um papel principal e positivo num SNS - Serviço Nacional de Saúde, com eficiência e rapidez na troca de informações e consequente melhoria no atendimento ao paciente [14].

As metodologias de representação, gravação e difusão de dados médicos e clínicos, ainda assim tem que ser alvo de diversos estudos de modo a se encontrar um grande número de projetos e iniciativas que apoiem esta evolução, bem como existir o interesse de outras partes interessadas, como organizações governamentais e académicas. É altamente recomendado que se definam normas e regras, bem como uma boa comunicação com mensagens consistentes e seguras [14].

Como já visto, o volume crescente de diferentes tipos dados pode também levar a falsas conclusões por parte dos técnicos, daí a necessidade crescente da implementação deste tipo de plataformas de monitorização inteligentes.

Um exemplo de sucesso de uma plataforma de Inteligência Artificial, desenvolvida pela Universidade do Minho em colaboração com o Centro Hospitalar do Porto, é a AIDA - Agência para a Integração, Difusão e Arquivo de Informação Médica. Esta plataforma veio solucionar parte dos problemas existentes a nível da integração, difusão e armazenamento de informação diversificada.

De forma sucinta, a AIDA é uma agência que tem por base o uso de agentes inteligentes que permite a comunicação com os diferentes sistemas existentes. Assim permite enviar e receber informação hospitalar, gerir, guardar e responder a pedidos de informação de terceiros. Tem associada uma diversidade de sistemas que têm como objetivo principal integrar, difundir e arquivar grandes volumes de informação de diversos sistemas hospitalares [15].

É então associado a este cenário e a este tipo de estruturas como a AIDA que surge a necessidade de utilização de sistemas de monitorização em com-

binacão com estas estrutura tendo como objetivo o controlo e prevençãõ de falhas em máquinas integradas num ambiente hospitalar. Em resumo, estes sistemas permitem aos administradores e gestores saber instantaneamente se os recursos em utilização por parte de uma máquina ou conjunto de máquinas estão operacionais e/ou funcionais. Para conseguir o pretendido, a gestão dos sistemas é conseguida com o controlo dos recursos, gestão de serviços, monitorização a tempo inteiro, produção de alertas e relatórios relativos a anomalias e estado dos sistemas [11]. Assim, a base dos sistemas de monitorização e gestão de falhas, inclui a recolha de informações referentes a hardware, software e comunicações dos equipamentos, bem como de dados acerca do seu estado e desempenho.

As principais questões a ter em conta quando se lida com sistemas, são a disponibilidade, a fiabilidade e a sustentabilidade do equipamento a ser monitorizado. Pretende-se com a constante monitorização, que uma falha associada a um determinado sistema seja adiada ou até mesmo evitada [16]. O principal objetivo será prever uma falha com a monitorização de vários parâmetros e assim, agendar uma manutenção economicamente viável aquando da identificação de uma potencial falha[16].

Para que a taxa de sucesso verificado num sistema de manutenção e monitorização seja grande é necessário que exista inteligência associada a este tipo de plataformas e, para tal, a temática da extração do conhecimento tem um papel crucial. Ao longo de todo o processo de controlo, milhares de dados deverão ser guardados e estruturados numa base de dados devidamente pensada o organizada, para que resulte uma fácil e funcional pesquisa. Deste modo, através do estudo e cruzamento de diversos parâmetros, conseguir-se-á, de forma automatizada, uma solução avançada para a resolução de problemas e previsão de falhas com vista a estruturar antecipadamente possíveis cenários futuros [16].

Uma plataforma que possua uma monitorização inteligente associada terá então a capacidade de evolução com novo conhecimento, o qual poderá ser obtido através de situações recentes e processos de resolução que vão surgindo.

Uma outra área em foco, com grande ligação à temática desde projeto

e com importância significativa para a funcionalidade desta aplicação, é o mundo web, o mundo da Internet. Estamos numa era em que a Internet faz parte do nosso cotidiano, sendo indispensável para tudo o que fazemos. As suas evoluções refletem-se na sociedade e a investigação científica caminha lado a lado com as suas progressões. Tem-se vindo a efetuar diversos estudos e a publicar linhas orientadoras que justificam que a Internet permite uma interoperabilidade global e eficaz. Em traços gerais, o funcionamento via *Web* pode ser definido como um estabelecimento de princípios e regras que são seguidos de forma rigorosa e disciplinada, permitindo a ligação entre sistemas de forma correta e sem erros. O acesso à informação é, assim, efetuado de uma forma simplificada e ao mesmo tempo eficaz, garantindo aos utilizadores qualidade na informação e gestão de diversas situações [17]. Assim esta plataforma faz uso do mundo web de forma a integrar os seus constituintes e, deste modo, interligar sistemas que se pretendem monitorizar, bem como aos seus utilizadores, em qualquer lado, com partilha de informação para gestão da mesma. Poderá, portanto, prevenir e corrigir situações anómalas de forma aberta numa perspetiva de comunicação global.

Resumindo, e tendo em conta a problemática acima contextualizada, é possível construir um sistema capaz de desenvolver a gestão de informação clínica em prol da evolução, apoiando os profissionais de saúde na resolução e prevenção dos problemas da instituição, bem como dos pacientes, garantindo exatidão, qualidade, velocidade, eficiência, facilidade de acesso e baixos custos.

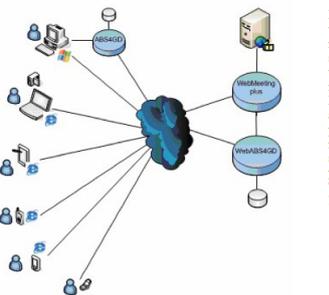


Figura 1.1: Arquitetura de um sistema com interação, comunicação e gestão através da Internet[1]



## 1.2 Estrutura da Dissertação

Esta dissertação divide-se em sete capítulos, os quais se encontram estruturados de forma simples, de modo a demonstrar e caracterizar os passos efetuados ao longo do trabalho desenvolvido.

No presente capítulo é feita uma introdução através de uma abordagem geral sobre a problemática em questão e o enquadramento teórico realizado, onde também se apresenta a estrutura da dissertação.

No capítulo dois, referente ao enquadramento teórico, são abordados temas relacionados com o projeto, fazendo a contextualização e a fundamentação teórica necessária à compreensão e ao conhecimento sobre o trabalho de investigação realizado. Neste capítulo são abordados temas de relevo para o enquadramento teórico, como os Sistemas de Informação ou a temática da Interoperabilidade.

O terceiro capítulo, referente à Visão de Produto, caracteriza e apresenta a plataforma construída de forma detalhada, onde se mostra as especificações do que vai ser desenvolvido.

O capítulo quatro, da Modelação do Produto, apresenta, como o próprio nome diz, a modelação da plataforma de monitorização. Neste capítulo converte-se a visão do produto numa planificação, de forma a criar uma espécie de guião durante a implementação.

O capítulo cinco refere-se ao desenvolvimento da aplicação, que expõe os resultados conseguidos depois da implementação, bem como as linguagens e tecnologias utilizadas.

O sexto capítulo é um caso de estudo e teste à aplicação, no qual se pretende simular o uso da mesma associado a um ambiente minimamente real.

No sétimo e último capítulo desta dissertação, são apresentadas as principais conclusões da elaboração deste projeto, bem como uma síntese do que pode ser ainda desenvolvido ser feito como trabalho futuro em prol da evolução do protótipo criado.

# Capítulo 2

## Trabalho Relacionado

Neste capítulo, como o próprio nome indica, está apresentado o trabalho relacionado com o projeto, onde será feita uma abordagem e um enquadramento teórico dos temas associados à aplicação desenvolvida. São eles, os Sistemas de Informação, a Interoperabilidade e a Descoberta de Conhecimento em Bases de Dados.

### 2.1 Sistemas de Informação

Mais do que um sistema de monitorização, a estrutura desenvolvida é, na prática, um sistema de informação, permitindo aos utilizadores uma troca de informação e dados, assim como uma interação entre estes e máquinas.

Nos últimos anos, os Sistemas de Informação em unidades de saúde têm vindo a ganhar uma enorme relevância e têm crescido em qualidade e quantidade. Com este crescimento de informação, torna-se necessário discernir o que é informação relevante e o que não é [18].

Torna-se, portanto, importante perceber o conceito de um sistema de informação, as suas principais características e funcionalidades, e ainda os diferentes tipos de sistemas de informação existentes.

Um sistema de informação é um termo que, atualmente, transporta associado a si um conjunto de definições, que diferem consoante o campo de aplicação do próprio sistema. Será então correto primeiramente clarificar o

que é a Informação e o que é um Sistema e, só depois de integrar o significado destes termos associados a um contexto específico, perceber o que é um sistema de informação.

## Informação

Na definição de informação é necessário ter em conta a diferença que existe entre o conceito de dados (*data*, em inglês) e o conceito de informação. Senão vejamos:

Dados podem ser definidos como um fator bruto que como um número, ou algo como uma data, ou uma medida. Por outro lado, informação refere-se aos dados que já sofreram um processamento e que são significativos. Isto requer um processo que seja usado para produzir informação, ou seja, que envolva recolha de dados, que posteriormente são submetidos a um processo de transformação de modo a criar informação [7]. Um exemplo, como se verá no segmento deste documento, acontece com a unidade de processamento que, a partir de dados brutos, consegue fornecer informação útil ao utilizador.

Apesar de a informação ser um recurso importante para organizações e instituições, nem toda a informação é considerada útil. As diferenças entre informação boa e má podem ser identificadas pelos atributos da qualidade da informação, que podem estar relacionados com o *timing*, conteúdo e forma da mesma.

## Sistema

Um sistema pode ser definido como o conjunto de componentes que trabalham em conjunto por um objetivo comum que, no caso, é receber *inputs* e transformar os mesmos em *outputs*. Alguns sistemas, constituídos por subsistemas com sub-objetivos não podem trabalhar isoladamente, mas sim em função do objetivo final do sistema [7]. Por exemplo, o sistema desenvolvido neste projeto é constituído por diferentes unidades que, unidas, constituem um objetivo final, de forma a monitorizar o desempenho de uma máquina. Para tal, é necessário que exista um mecanismo que devolva um *feedback* e, ao mesmo tempo, uma determinada ação deve ser exercida para solucionar

qualquer problema que ocorra, garantindo assim que o sistema está dentro dos parâmetros pretendidos.

### Sistemas de Informação

Agora que foi clarificado o conceito de informação, e o de um sistema, pode definir-se o que é um Sistema de Informação. Trata-se, portanto, de um sistema onde a informação deve ser capturada, representada, armazenada e processada no contexto da organização, das pessoas e da tecnologia, para resolução de diferentes problemas. O papel destes sistemas é assim fornecer informação para monitorização, o que permita tomar decisões controladas, uma vez que só existe controlo nas decisões se houver também conhecimento acerca das necessidades de um ambiente.

Estes podem ser vistos como um grupo de componentes interligados que trabalham em coletivo para produzir entradas, processamentos, saídas, armazenamentos e controlar ações, de modo a converter dados em produtos de informação que possam ser usados para previsão, planeamento, controlo, coordenação, tomadas de decisão e atividades operacionais. No que diz respeito aos componentes associados ao sistema, podemos destacar as pessoas, hardware, software, comunicações e dados. Pessoas engloba os utilizadores e os programadores da aplicação, bem como aqueles que colaboram na manutenção do sistema, como gestores e técnicos. Em relação aos recursos de hardware, destacam-se os computadores e outros componentes. Os recursos de software englobam os programas que são executados pelas máquinas, enquanto as comunicações englobam redes e Internet bem como todos os componentes de hardware que são necessários. Por fim, os recursos de dados dizem respeito aos dados que uma organização tem acesso, como bases de dados [7].

Os sistemas de informação podem dividir-se em dois grandes grupos:[7]

- OIS - Sistemas de Informação de Operações, os quais estão mais focalizados para controlo de processos, processamento de transações e comunicações;

- MIS - Sistemas de Informação de Gestão, mais focalizados em fornecer suporte para a tomada de decisão no campo da gestão.

Em comparação com o sistema realizado, percebe-se, assim, que este combina as definições apresentadas, num contexto prático de utilização, pois faz uso de dados, transformando os mesmos em informação para o utilizador receber e analisar de modo a controlar processos e comunicações ainda assim auxiliar na tomada de decisões que se poderão revelar úteis no campo da gestão

### **Segurança nos Sistemas de Informação**

Ainda associado a sistemas de informação e não menos importante na relação com o tema do trabalho desenvolvido, surge o sub-tema da segurança que, nos sistemas como o implementado, assume um papel de relevo. Assim, na Tabela 2.1) são apresentadas as ameaças mais prováveis de ocorrerem num sistema com esta tipologia, bem como os métodos de proteção contra essas mesmas ameaças. Como se verifica, existe maior ênfase dada às áreas dos vírus de computadores e ameaças ao serviços via Internet. Os controlos em sistemas de informação são baseados em dois princípios: o da necessidade de garantir precisão nos dados mantidos pela organização, e o da necessidade de os proteger contra perdas ou danos [7].

Tabela 2.1: Resumo da segurança associada a um Sistema de Informação, com as ameaças e possíveis soluções [7].

| <b>Ameaças de Segurança</b> | <b>Redução das Ameaças</b> | <b>Tipos de Controlo</b>            | <b>Técnicas de Controlo</b>   |
|-----------------------------|----------------------------|-------------------------------------|-------------------------------|
| Acidentes                   | Contenção                  | Proteção Física                     | Segurança Policial            |
| Desastres Naturais          | Detenção                   | Controlos biométricos               | Passwords                     |
| Sabotagem                   | Ocultação                  | Controlos por telecomunicações      | Encriptação                   |
| Vandalismo                  | Recuperação                | Controlo de falhas                  | Procedimentos organizacionais |
| Roubo                       |                            | Auditorias                          | Validação de Utilizador       |
| Uso não autorizado          |                            | Detetar e prever infeções por vírus | Backups                       |
| Vírus Informáticos          |                            |                                     |                               |

## 2.2 Interoperabilidade

Com o aumento das redes e do poder computacional, tanto de servidores como de dispositivos dessas mesmas redes, são enormes as quantidades de tarefas executadas em tão curto espaço de tempo. Dessa forma, torna-se impraticável a execução de monitorizações manuais num tão grande número de servidores e dispositivos de rede. Como consequência, surge a necessidade de implementação de um sistema de monitorização automatizado para aplicações e equipamentos de uma rede, que não só divulgue imediatamente o estado desses equipamentos, como também consiga, caso possível, corrigir determinados problemas automaticamente, sem a intervenção dos respetivos administradores [19].

Por norma, um sistema de monitorização encontra-se ligado e sincronizado a uma rede de computadores. As redes em questão apresentam diversos

níveis de complexidade, podendo ter desde apenas um par de computadores, como possuir dezenas de máquinas ligadas com diferentes sistemas operativos e correndo diversos processos.

Segundo a definição de interoperabilidade, um sistema tem que ter a capacidade de comunicar com outro sistema de forma transparente, com padrões abertos ou na mesma linguagem, e utilizando protocolos comuns. [20]

Considerando que nas redes e suas respectivas máquinas existe quase sempre um conjunto alargado de aplicações e sistemas que geram eventos, depressa se conclui que, devido à existência de eventos de variadas origens, existirá a dificuldade de interação com a estrutura de monitorização. [21] Assim, esta heterogeneidade existente ao nível dos sistemas é um contra para a interoperabilidade que se pretenda que exista.

Em síntese, se considerarmos cada entidade do sistema como um agente, que seja pró-ativo e reativo, existem soluções para contrariar esta heterogeneidade, de forma a atingir uma interoperabilidade completa. Estas podem enumerar-se em: [22]

- Interface de agente comum, que permite o básico para uma interoperabilidade mínima e funcional para qualquer linguagem de programação segundo as especificações IEEE-FIPA-Institute of Electrical and Electronics Engineers. Possibilita também aos agentes interagir em diferentes localizações com diferentes *middlewares*<sup>1</sup>; (Figura 2.1)
- Conjunto de modelos que permitem aos diferentes *middlewares* suportar agentes desenvolvidos em diferentes linguagens e arquiteturas;
- Método baseado em agentes com múltiplos códigos, bem como um agente comum com capacidade de descodificação que permite a interoperabilidade entre *middlewares* que não suportam a mesma linguagem de programação.

---

<sup>1</sup>Mediador existente entre os sistemas operacionais e as aplicações executadas;

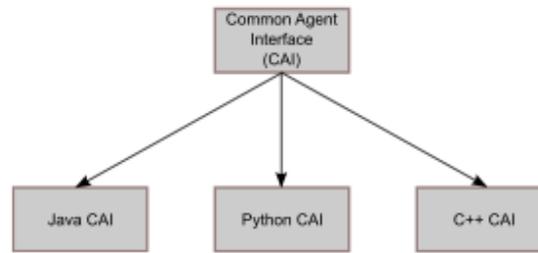


Figura 2.1: Esquema de uma interface de agente comum [22]

Como já visto, para que se verifique interoperabilidade, é necessário um elevado grau de interacção entre os ambientes existentes e os próprios agentes. De facto, esta situação é fortemente dificultada pelos diferentes tipos de agentes existentes ao nível dos *middlewares*, bem como pelas diferentes linguagens de programação e ainda por diferentes arquiteturas.

Na prática, dois sistemas passam a ser interoperáveis se um agente móvel conseguir interagir e comunicar com outros agentes, quer localmente quer de forma remota, ou se um agente de um dos sistemas tiver capacidade de migrar entre sistemas e conseguir executar alguma operação em cada um deles [22].

Em Cucurull et al. [22], apresentam quatro diferentes áreas que podem ser conjugadas para se conseguir uma máxima interoperabilidade. Vejámos:

- Área de Execução – A existência de diferentes linguagens de programação e arquiteturas de computadores causam dificuldades na partilha de um único formato de código executável;
- Área de *Middleware* – Cada sistema de agentes móveis usa uma *middleware* e uma API específicas com um sistema de controlo de agentes. Por conseguinte, agentes com diferentes *middleware* são normalmente incompatíveis.
- Área de Comunicação – Diferentes *middleware* têm diferentes mecanismos de comunicação, cada um com as suas próprias estruturas de

mensagens e métodos de entrega, o que dificulta a comunicação de agentes.

- Área de Mobilidade – A migração de um agente implica o cumprimento das etapas do processo de migração, e do conjunto de mensagens trocadas, bem como da informação incluída neles. De outra forma, a migração não seria possível;

A definição de interoperabilidade é um assunto que, atualmente, transporta alguma subjetividade associada, uma vez que a conotação dada a este termo varia consoante o cenário em questão. Deste modo, a definição para a interoperabilidade pretendida por uma organização de cuidados de saúde será substancialmente diferente da definição para uma organização não ligada à saúde. As diferentes definições encontradas atualmente para interoperabilidade são dadas sobretudo por organizações de saúde, associações profissionais e agências governamentais. Ainda assim, em comum às referidas, existem três tipos principais de interoperabilidade: [23]

- Interoperabilidade Técnica;
- Interoperabilidade Semântica;
- Interoperabilidade de Processo;

Hoje em dia, em todo o mundo, organizações e governos trabalham no sentido único de melhorar a tecnologia de informação na saúde. Um uso inadequado das tecnologias de informação e investimentos desmedidos nas mesmas, levam a um aumento considerável nos custos dos cuidados de saúde, o que faz com que seja necessário a existência de infra-estruturas organizadas e sobretudo interoperáveis.

Existem muitos planos de saúde relacionados com a definição de interoperabilidade, como a interoperabilidade dentro de um hospital ou clínica. Um plano simples relaciona-se com a interoperabilidade que deve existir entre os vários sistemas auxiliares. Por exemplo como do sistema de informação laboratorial - LIS ao sistema de registos de saúde - EHR, onde a necessidade de

conversão de informações a partir das linguagens nativas de um outro sistema apresenta uma outra dimensão da interoperabilidade [23]. A interoperabilidade também se torna termo chave quando se trata da troca de informação de saúde entre diferentes instituições, onde os sistemas existentes ao nível de cada uma, muitas vezes, não possuem a mesma arquitetura nem fazem uso da mesma linguagem.

Estudos realizados para encontrar a melhor definição para o termo interoperabilidade, a partir de informação recolhida de diferentes organizações e associações, comprovaram que as organizações de saúde foram as que mais contribuíram e as que reuniram maior consenso do que seria a definição ideal de interoperabilidade[23]. Comprova-se assim a importância deste conceito a nível hospitalar. Interoperabilidade nos cuidados de saúde pode ser definida como a capacidade existente nos sistemas para se mover com facilidade dados clínicos entre diferentes lugares. Existindo interoperabilidade associadas a essas instalações torna-se possível reduzir custos e oferecer meios mais eficientes e poderosos na assistência ao paciente, em especial no apoio á decisão e na resolução de problemas[24].

Ainda como resultado do estudo feito, e apesar de existir grande concordância nas definições dadas para o termo interoperabilidade, também se verificam algumas diferenças nas opiniões. Todos os intervenientes basearam as suas definições na definição técnica para o termo, assumindo a interoperabilidade como uma questão de simples conectividade, desde que exista capacidade de garantir que a mensagem foi trocada (enviada ou recebida) no formato correto. No entanto, esta definição não inclui a noção semântica, que se preocupa com a preservação do conteúdo. Assim, segundo interoperabilidade técnica, se uma mensagem chegar ao local de receção, não teria importância se o destinatário compreendeu ou não o conteúdo, apenas se a mesma chegou e no formato correcto.

Segundo *Angelo Rossi-Mori, et al.* em “*Computer-based handling of clinical information*”, a interoperabilidade técnica é o primeiro objetivo a atingir, enquanto que a interoperabilidade semântica está relacionada com as funções do sistema como organização da informação , atualizações de conteúdos, bem como ferramentas de optimização, como apontadores para tabelas de bases

de dados. A interoperabilidade de processo oferece a capacidade de reconhecer o conteúdo dos dados transmitidos e colocá-los num formato compatível para leitura no sistema de chegada [25].

Segundo a *IEEE Standard Computer Glossary*, num contexto global, interoperabilidade refere-se à habilidade de dois ou mais sistemas ou componentes trocarem informação. de modo a utilizar essa mesma informação trocada [23]. Para esta associação, a interoperabilidade é um dos conceitos mais importantes a adotar e a ter em conta na implementação das tecnologias de informação.

Assim, em relação com a área da saúde, a capacidade de trocar informação não significa apenas que os sistemas tenham capacidade de comunicar uns com os outros, mas sim de partilha das mesmas terminologias, definições e linguagens, de forma a colocar a informação o mais distribuído possível entre todos os sistemas hospitalares [23].

Resumindo:

**Interoperabilidade Técnica:** é o conceito usado para definir a forma mais básica de interoperabilidade baseada na camada de hardware e pode ser definido como a capacidade de dois sistemas diferentes trocarem dados de forma eficiente. Foca-se especificamente no transporte de dados e não no conteúdo, ou seja, envolve a transmissão e receção de informação que pode ser usada por um utilizador mas não pode ser tratada posteriormente numa semântica equivalente por um software [23].

**Interoperabilidade Semântica:** de forma a retirar o máximo partido da informação partilhada entre sistemas, desde o uso de aplicações de apoio à decisão inteligente, é necessário um nível mais profundo de interoperabilidade. É neste contexto que surge a interoperabilidade semântica que pode ser definida como a capacidade dos sistemas para entender a informação partilhada. Trata-se de um conceito multi-nível dependente da terminologia dos dados, dos arquétipos, bem como dos *templates* usados, quer pelo sistema de envio, quer pelo sistema que recebe a informação. É de notar que, quanto maior for o nível de inte-

roperabilidade semântica associado ao software, menos processamento humano é necessário. Esta situação oferece vantagens, como diminuição do erro associado à introdução e análise de dados por parte do utilizador. Em contrapartida fica mais sujeito à intrusão de informação e dados que poderão não ser importantes[23].

**Interoperabilidade de Processo:** é definido como um conceito emergente no campo da interoperabilidade que tem sido identificado como um requisito essencial para a implementação de um sistema de sucesso de acordo com as exigências existentes, atualmente, no mundo do trabalho. É visto como uma chave no que visa a melhoria da qualidade e segurança, lidando principalmente com os métodos para uma integração ideal dos sistemas informáticos em ambientes de trabalho reais. Assim, interoperabilidade de processo oferece as seguintes funcionalidades/vantagens: [23]

- Interação utilizador-sistema eficiente, com utilidade e *user friendly*;
- Fluxo de dados e apresentação compatível com diferentes definições de trabalho;
- Ambiente de trabalho estruturado;
- Definição da função do utilizador;
- Eficácia demonstrada em casos reais.

Concluindo o conceito de interoperabilidade permite ao utilizador entender, em termos gerais, como os sistemas comunicam e interagem entre si, e a importância desta mesma interação. Esta pode referir-se a interações inter-institucionais, intra-institucionais, bem como interações intra-sistemas. Interoperabilidade existe quando um ou mais sistemas são considerados numa relação um com o outro, e não sendo correto abordar como algo físico. Pode variar desde inexistente (entre dois ou mais sistemas), a abrangente e completa, permitindo a total transparência e completa simultaneidade entre os dois sistemas[23].

Em forma de síntese, verifica-se uma interoperabilidade superior quando assistimos a uma relação entre três principais definições - técnica, semântica e processo[23].

Tabela 2.2: Níveis de interoperabilidade segundo Mykkanen and Tuomainen (2008)[8]

| Level | Designation                       | Meaning   |
|-------|-----------------------------------|---|
| 7     | Application life cycle interfaces | The life cycle of the application, including integration and development methods          |
| 6     | Functional reference model        | The domain-specific information or functional model or assumptions about the used methods |
| 5     | Semantics                         | The meaning of the defined interface elements   |
| 4     | Functional interfaces             | The defined functionality and information   |
| 3     | Application infrastructure        | The integration points in the distribution architecture of the participating applications |
| 2     | Technical infrastructure          | The infrastructure for supporting the interface and communication technologies            |
| 1     | Technical interfaces              | The technologies used in the interfaces and implementations                               |

Em síntese, uma Unidade de Saúde é constituída por uma conjunto enorme de aplicações que falam linguagens diferentes, que são customizadas de maneira distinta pelos seus criadores sendo assim essencial uma comunicação concreta e eficaz entre sistemas - Interoperabilidade, para que não se perca informação nem se passe a informação errada [26][27].

*Together, these three types of interoperability are all required to the consistent and timely achievement of what has come to be called “Best Practice.”*[23]

## 2.3 Descoberta de conhecimento em Bases De Dados

De acordo com a evolução das Tecnologias de Informação, assiste-se cada vez mais a um considerável aumento da quantidade de dados utilizados e trocados por parte dos sistemas. Assim, as Bases de Dados assumem um papel preponderante e de grande utilidade para o armazenamento, organização e utilização desses mesmos dados. Como resultado desta evolução aliada a alguma facilidade de uso das TI, associações e organizações têm verificado um crescimento exponencial dos seus dados. Este volume de dados e a sua complexidade transporta consigo a dificuldade de análise por métodos estatísticos clássicos, bem como pelo ser humano. Tal como se poderá vir a perceber em capítulos mais adiantados deste projeto, a aplicação desenvolvida tem como elo de ligação entre as diferentes estruturas criadas uma base de dados, assumindo um papel fundamental e decisivo em todo o seu funcionamento.

No que diz respeito à análise e estudo dos dados registados, apenas uma pequena parte dos mesmos são usados como base e instrumento de apoio à tomada de decisão, ficando os restantes guardados para posterior uso, se necessário. [28]

Neste cenário convicto de que as Bases de Dados são ferramentas poderosas que, quando submetidas a uma análise, estudo e compreensão de dados, permitem que, através de ferramentas de representação e extracção de conhecimento e de DM, dados em bruto sejam potencialmente úteis e compreensíveis.

Este processo do qual se retira informações, ou seja, conhecimento a partir de um conjunto de dados armazenados, recorre a técnicas de *Data Mining*. Nestas são aplicados algoritmos de extração de padrões, incorporando conhecimentos de domínio de aplicação, através da interpretação adequada dos resultados [29]

Segundo Fayyad et al., todo o processo DCBD é um processo interativo e iterativo que passa por vários estados. São eles:[30]

- Definição do domínio de aplicação;
- Criação de um conjunto de dados alvo;
- Limpeza e pré-processamento de dados;
- Redução e projeção de dados;
- Escolha e aplicação do método/algoritmo de DM mais adequado;
- Interpretação dos padrões obtidos;
- Utilização e/ou documentação do conhecimento obtido.

Como se observa pela Figura 2.2, de acordo com *Fayyad et al.*, as etapas anteriormente enumeradas podem ainda ser inseridas em etapas mais globais, tais como: [31]

- Seleção de Dados - De acordo com os objetivos do processo, faz-se a recolha dos dados que se considerem úteis e excluem-se os que não são relevantes para o processo de descoberta de conhecimento;
- Tratamento dos Dados - Realizam-se as operações de limpeza. É feito um processamento que engloba tratamento de dados errados, distribuição de dados não uniformes e outras incorreções detetadas mediante a verificação de inconsistências;
- Pré-processamento dos Dados - Feita de forma a facilitar a análise, onde os valores contínuos são transformados em valores discretos. Os dados passam a ser agrupados por classes;
- Aplicação de algoritmos de *Data Mining* - Seleção de métodos e aplicação de algoritmos e técnicas para extração de padrões de dados;
- Interpretação e validação dos resultados - Visualização do conhecimento obtido e respetiva interpretação e avaliação, sendo possível retomar o processo em qualquer uma das anteriores etapas para nova iteração.

Resumindo, o crescimento desmesurado da quantidade de dados armazenados nos computadores atuais, consequência das TI, motivou o aparecimento da Descoberta de Conhecimento em Bases de Dados, visando a automatização da extração de conhecimento. A motivação prende-se com o fato de, paralelamente à dimensão das bases de dados, crescer também a complexidade das relações entre os dados. Estes dois fatores são deveras limitadores da capacidade humana de extração de conhecimento útil das bases de dados[31].

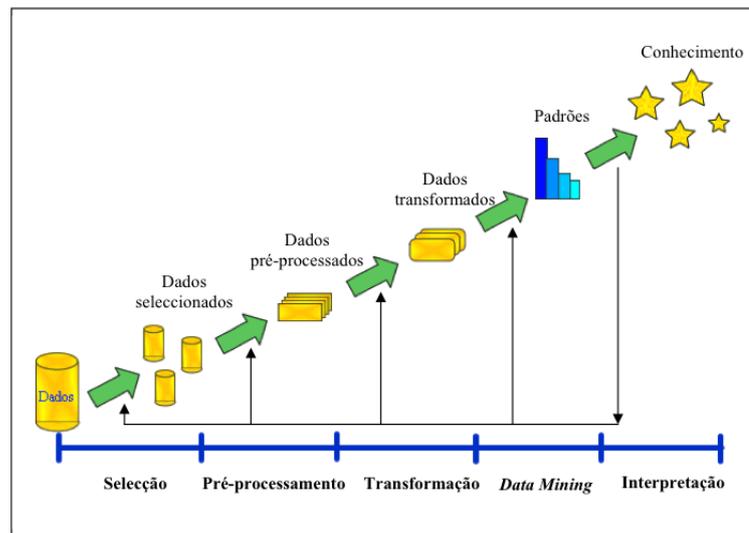


Figura 2.2: Processo DCBD segundo Fayyad et al. [31]

## 2.4 *Data Mining*

Inserido no contexto temático do trabalho desenvolvido para esta dissertação está o conceito de *Data Mining*. Não estando, de certa forma, desenvolvido e aplicado neste trabalho de descoberta de conhecimento em Bases de Dados, espera-se que no futuro este conceito esteja verdadeiramente implementado e seja determinante para o correto e inteligente funcionamento desta aplicação, daí haver uma secção direcionada apenas para este assunto.

Devido ao facilitismo verificado nos dias de hoje para reunir, armazenar, processar e partilhar dados, tem-se assistido a um crescimento exponencial da quantidade de dados armazenados que podem auxiliar na tomada de decisão, o que levou ao interesse nas áreas de *Data Mining* (Turban et al., 2007). [30]

Mas o que é *Data Mining*? Qual a sua importância?

Em Cruz and Cortez [30], *Data Mining* é um processo para extrair um conhecimento aprofundado a partir de dados em bruto/originais, ou seja, dados que ainda não sofreram qualquer tipo de manipulação. Já em [32] *Data Mining* é um conceito definido como o processo de descoberta de padrões em dados, onde este deverá ser automático, ou (como normalmente usado) semi-automático. Os padrões descobertos devem ser significativos na medida em que tragam algo vantajoso, como por exemplo, uma solução economicamente viável.

Resumindo, *Data Mining* refere-se apenas a uma pequena parte (10% a 15%) de todo o processo de DCBD - Descoberta de Conhecimento em Bases de Dados que visa a aplicação de algoritmos com vista à descoberta de padrões em dados já pré-processados. É de referir que o termo *Data Mining* é mais simples e sonante que DCBD, sendo que na prática ambos os termos tendem a ser utilizados como sinónimos, o que não é de todo correto. É uma etapa crucial de todo o processo, onde o seu sucesso depende claramente do bom trabalho realizado nas etapas precedentes[30].

As aplicações de *Data Mining* dividem-se pelas mais variadas áreas, embora apresente mais ênfase em algumas, como é o caso das Tecnologias de Informação. Além desta, as áreas de maior destaque são:

### ***Customer Relationship Management (CRM)***

*Data Mining* permite fornecer conhecimento a uma instituição sobre os seus clientes e suas necessidades, possibilitando assim a criação de

novos e melhores serviços, bem como de produtos de acordo com o seu mercado [33];

### **Suporte a decisão**

A tomada de decisão tem influência direta no futuro e a melhor decisão tomada tem vantagens para uma organização, quer seja empresa, organização ou um ambiente hospitalar. O *Data Mining* poderá fazer a diferença fornecendo melhores previsões e identificando padrões, entre outros [34];

### **Finanças**

É usado para deteção de fraudes, análises de créditos, previsões, etc. [34];

### **Investigação Científica**

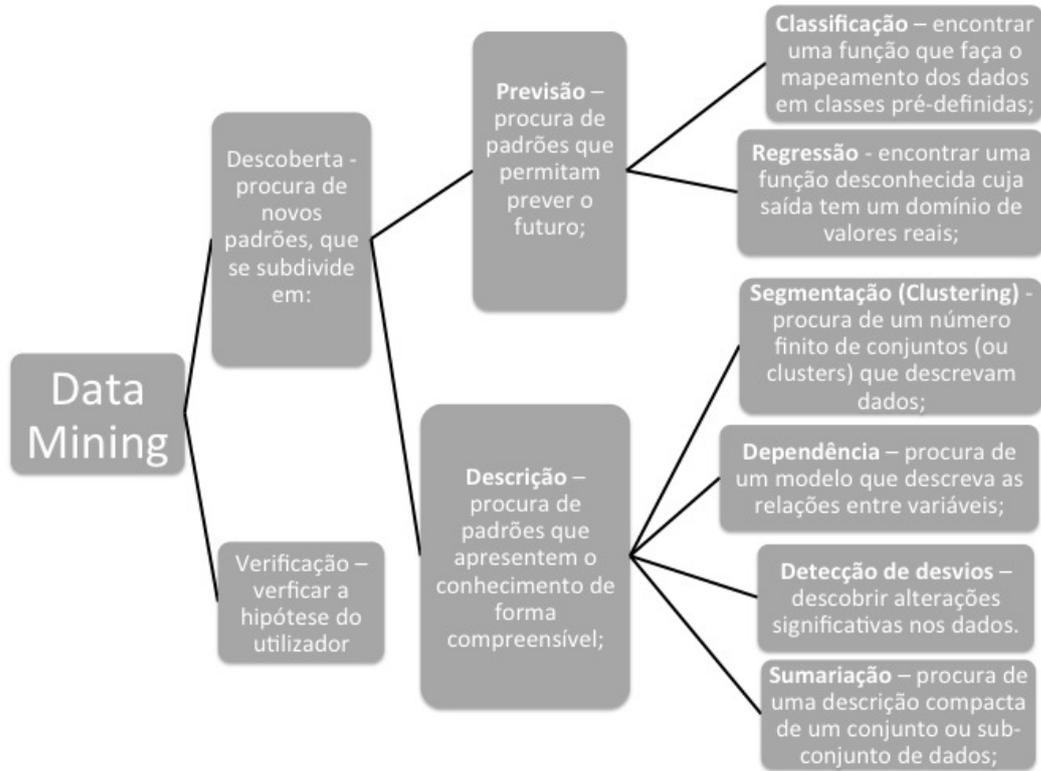
Com destaque em aplicações médicas, na fase de diagnóstico, na identificação de melhores prescrições, bem como na pesquisa de novas formas de tratamento [31];

### **Outras aplicações**

Outras aplicações do *Data Mining* passam em parte pelos dispositivos tecnológicos e sistemas informáticos, em exemplos como o *Web Mining*, *Biblio Mining*, e descoberta de conhecimento em Bases de Dados Multimédia [35].

### **Métodos de *Data Mining***

No esquema apresentado a seguir, Figura 2.3 está representada a hierarquia do conceito de *Data Mining*. Percebe-se então que neste se pode distinguir dois objetivos, a **Verificação** e a **Descoberta**. Para o objetivo da **Previsão**, os problemas são tratados como pertencentes a cada uma das classes mostradas. Para o objetivo de **Descrição** existem os métodos esquematizados, que são **Segmentação (clustering)**, **Sumariação**, **Dependência** e **Deteção de desvios** [31].

Figura 2.3: Esquema sucinto do termo *Data Mining*

# Capítulo 3

## Visão do Produto

Os capítulos anteriores demonstraram e deram a perceber a importância e a necessidade da aplicação de uma estrutura de monitorização em sistemas de tecnologias de informação e, principalmente, o impacto positivo associado a um ambiente hospitalar, pelas suas características. Agora, no capítulo que se segue, será mostrada e apresentada a ideia para uma plataforma de monitorização com as capacidades que respondam às exigências encontradas atualmente.

### 3.1 Visão Global

A plataforma de monitorização desenvolvida tem que ser um aplicativo acessível e fácil de usar que forneça às instituições uma ajuda fundamental na monitorização das suas máquinas e dos seus processos, bem como nas comunicações efectuadas via *Web*.

Desta forma, permite uma visão integral dos ambientes de trabalho e ajuda a detetar problemas antes que estes possam efetivamente causar falhas nos sistemas. Também monitoriza o desempenho dos seus componentes e envia alertas quando surgem problemas. Tem a capacidade de criar uma espécie de diagnóstico através de relatórios integrados, de modo a solucionar e ajudar na resolução de problemas.

Como é possível observar no esquema da Figura 3.1, esta plataforma de monitorização permite a vigia de sistemas através do uso da Internet. Assim, os utilizadores podem visualizar as suas máquinas, controlar os respetivos parâmetros e receber alertas, independentemente de onde se encontram.

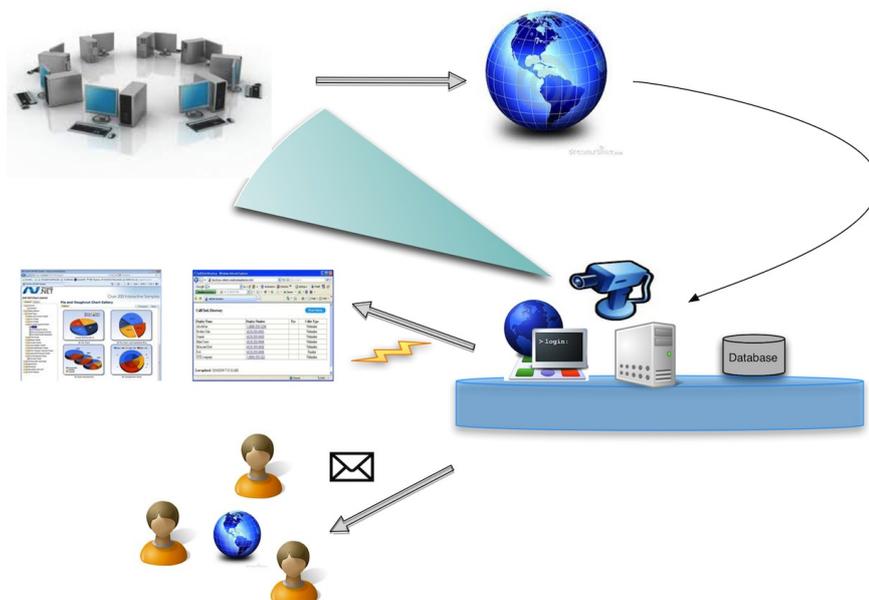


Figura 3.1: Esquema representativo do funcionamento global da plataforma de monitorização

Esta plataforma, no seu funcionamento, necessita de seguir as seguintes fases:

### **Recolha de Informação/Dados**

A informação é reunida, sendo uma fase automática e constante;

### **Armazenamento da Informação/Dados**

É necessário armazenar os dados reunidos, o que implica que os mesmos sejam divididos em diferentes categorias e estejam relacionados entre si para que se consiga um fácil armazenamento e acesso;

### **Estudo e Verificação**

É necessária uma verificação constante dos dados armazenados, de

modo a que no seguimento de qualquer registo anómalo surja o envio de um alerta;

#### **Apresentação de Resultados/Informação**

A informação recolhida e os dados armazenados serão mostrados de forma organizada e perceptível ao utilizador através de uma interface gráfica. Poderá ser sob a forma de tabelas de dados e/ou gráficos estatísticos;

#### **Interação com a Informação Apresentada**

O utilizador irá observar e analisar a informação apresentada, com o objetivo de perceber o estado das máquinas monitorizadas e assim gerir essas mesmas informações do modo que lhe seja mais favorável.

Resumindo, esta aplicação permite uma visão integrada de diversos recursos que compõe o sistema, controlando a disponibilidade e desempenho, para que os utilizadores não sejam afetados. Estes podem, assim, aperceber-se do estado crítico dos recursos e serem alertados em tempo real.

## **3.2 Estrutura e Constituintes**

Nesta secção serão abordadas e explicadas cada uma das unidades individuais que compõe o sistema. Estas unidades, cada uma com as suas funções específicas, quando combinadas entre si, oferecem à estrutura desenvolvida todas as características que lhes são conhecidas. Na Figura 3.2 está representada a estrutura global da aplicação, onde pode ser visualizado cada um dos constituintes, assim como uma sumarização das funcionalidades associadas a cada um.

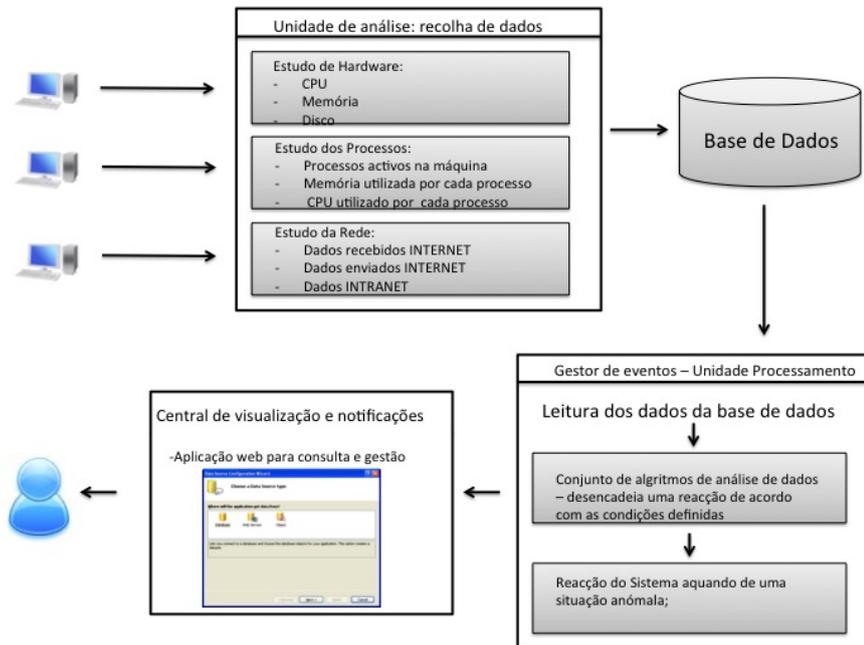


Figura 3.2: Estrutura da aplicação com os seus diferentes constituintes

### 3.2.1 Unidade de Análise

A unidade de análise é o ponto de partida de todo o processo de análise de dados, daí a origem do nome dado. Consiste numa arquitetura construída em linguagem C#, em combinação com o WMI[4] (arquitetura explicada numa outra fase deste documento), que tem como principal objetivo recolher os dados referentes às máquinas que se pretendem monitorizar, e escrever esses mesmos dados numa Base de Dados. A aplicação foi desenvolvida sob a forma de um ficheiro executável que funciona em ambiente *background*, não apresentando, portanto, qualquer incomodo para o utilizador. Desta forma, todos os dados que são considerados importantes e relevantes para um perfeito funcionamento da máquina são recolhidos e escritos na base de dados.

Os dados recolhidos classificam-se em quatro grandes grupos. Vejamos: Dados de identificação da máquina em questão, dados sobre os processos que estão a ser executados pela mesma, dados de hardware e dados de rede.

Quanto aos dados de identificação, optou-se por definir uma máquina, pelo seu MAC Address<sup>1</sup> e pelo IP<sup>2</sup> que lhe está atribuído. Relativamente aos processos, são recolhidos nome, memória ocupada e processador utilizado por cada processo em execução, no momento exato da análise.

Quanto ao hardware, são recolhidos dados sobre os discos rígidos, processador e memória RAM, tais como capacidades e percentagem de utilização para cada um deles. No que diz respeito aos dados de rede, são monitorizadas as trocas de dados feitas pela máquina, com o exterior, via Internet ou Intranet. Consegue-se, assim, perceber o tráfego de dados recebidos e enviados, bem como o número de pacotes de dados trocados.

---

<sup>1</sup>endereço físico associado e único, gravado em hardware[36];

<sup>2</sup>Internet Protocol;

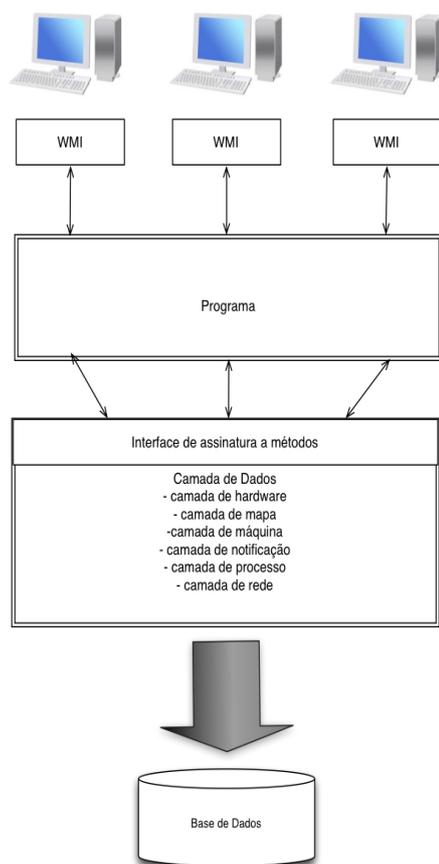


Figura 3.3: Esquema representativo da unidade de análise e seus diferentes constituintes

Como se vê na Figura 3.3, esta unidade foi desenvolvida e organizada sob um modelo "n camadas", arquitetura a qual será abordada e explicada com mais pormenor numa outra secção deste documento. Assim, esta unidade de análise oferece eficiência à plataforma desenvolvida, combinando velocidade de execução e fiabilidade na recolha dos dados.

### 3.2.2 Unidade de Processamento

Elaborada também em linguagem C#, a unidade de processamento pode ser vista como a unidade crucial e, de certa forma, a mais importante para o principal objetivo de funcionamento da plataforma de monitorização. Como

o nome indica, é através desta unidade que é feito o processamento dos dados. É uma aplicação de funcionamento contínuo e de constante vigia da informação existente na Base de Dados, que desencadeia uma determinada ação em caso de situação anómala (erro ou falha) nos sistemas que estão a ser monitorizados. Não será o termo mais correto, mas pode ser descrita como a unidade inteligente desta plataforma, onde está definido o “conhecimento” existente. Na prática é a parte da estrutura que tem definidas todas as condições, do que é considerado uma falha ou erro para o sistema.

Em caso de anomalia verificada, esta unidade, além de processar o envio de uma notificação para o utilizador, também cria um relatório com todas as informações acerca da falha verificada. A notificação desencadeada é transmitida ao utilizador via email. Um exemplo do tipo de condição para o qual esta unidade age, está relacionado com os processos. Caso um processo definido como crucial para o utilizador falhe em determinada máquina monitorizada, esta unidade avisaria de imediato o utilizador, como já referido, através de email. Ainda assim, convém referir que um processo é considerado crucial a partir do momento em que é selecionado pelo utilizador na aplicação *Web* para uma monitorização mais específica.

As condições definidas nesta unidade para que seja gerada uma notificação são:

- Para os Processos:
  - Falha de qualquer processo que seja definido como crucial por parte do utilizador;
  - Processos que, sendo de duração temporária, excedem o seu tempo de execução;
  - Utilização excessiva, ou seja, fora dos percentis definidos para os recursos, como processador e memória por parte de determinado processo.
  
- Hardware:
  - Se os valores para a memória e processador ultrapassarem um determinado valor médio mais um desvio padrão;

- Capacidades dos discos rígidos ultrapassarem 95 por cento de ocupação.
- Rede:
  - Se as comunicações via Internet não existirem durante um longo período de tempo, definido à posteriori.

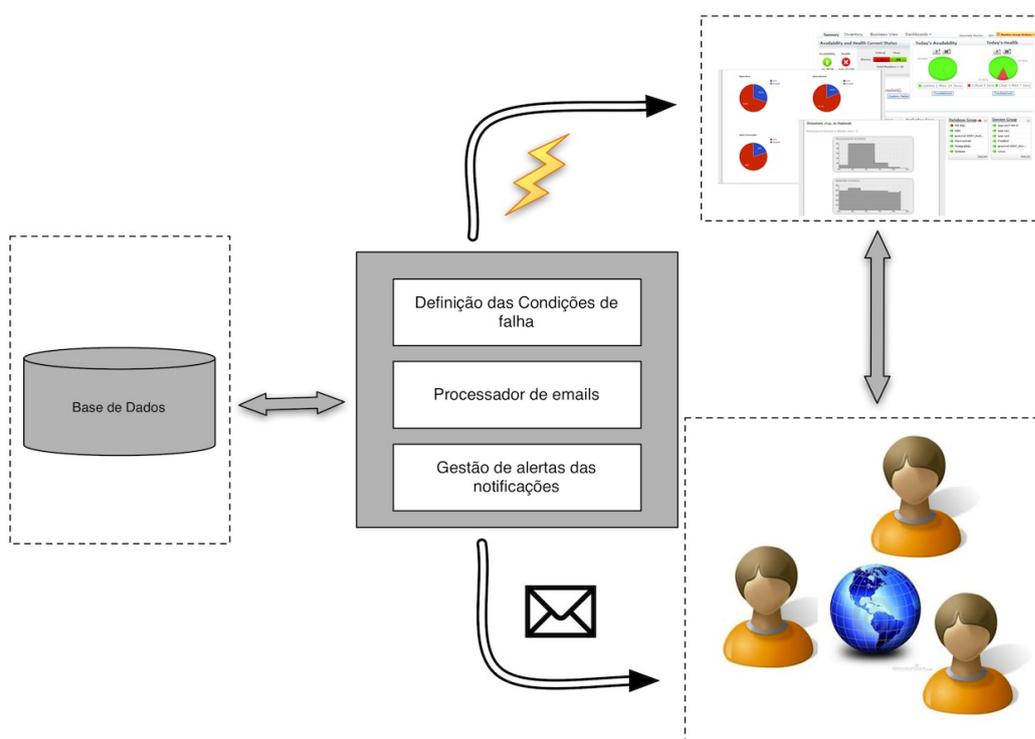


Figura 3.4: Esquema representativo da unidade de processamento e as suas funções.

### 3.2.3 Aplicação *Web*/Interface de Utilizador

É a parte visual/gráfica, e a única acessível ao utilizador, onde este tem real interação com a aplicação. Trata-se do ambiente principal, onde se pode fazer uso de todas as funcionalidades suportadas. Desenvolvida sob a forma de um *WebSite* e alojado num servidor, permite ao utilizador ter

acesso à plataforma de monitorização onde quer que se encontre. Consegue, portanto, informar-se em qualquer ocasião sobre as máquinas analisadas, tendo a possibilidade de resolver erros e anomalias em caso de necessidade, com a maior brevidade possível.

### 3.3 Características do Produto

Como já explicado e abordado anteriormente, a plataforma desenvolvida tem como finalidade principal a monitorização, de forma remota, de um conjunto de máquinas/computadores onde se pode visualizar todas as suas atividades. De forma a alcançar estes objetivos existe um conjunto de características associadas. Estas são:

- Recolha de dados simples e de forma eficiente por parte de uma unidade de análise;
- Processo de funcionamento em background sem incomodo para o utilizador;
- Monitorização a partir de qualquer lado, implicando apenas o acesso à Internet;
- Visualização em tempo real das máquinas que estão a ser monitorizadas, permitindo saber se se encontram ligadas ou desligadas;
- Para cada uma, possibilidade de:
  - Ver processos e todas as informações relativas aos mesmos;
  - Ver Dados de Hardware;
  - Ver Dados de Rede;
- Histórico do comportamento e desempenho das máquinas monitorizadas;
- Auxílio gráfico para melhor perceção;

- Unidade de processamento e controlo com central de notificações associada;
- Envio de notificações via e-mail para o utilizador aquando de uma situação anómala verificada;
- Oferece ao utilizador a possibilidade de agendar uma notificação específica;
- Possibilidade de leitura de relatórios detalhados sobre falhas registadas.

### 3.4 Síntese

O capítulo 3 permitiu representar uma visão do produto, para além de impulsionar e facilitar a transição para a fase seguinte deste projeto. Assim, percebeu-se quais as unidades e componentes que constituem a plataforma, os objetivos que se pretende atingir com esta aplicação. Este capítulo transmitiu também que este produto se concentra essencialmente na interação homem-máquina, como um meio para melhorar as atividades dos profissionais.

# Capítulo 4

## Modelação do Produto

### 4.1 Metodologias

Antes de iniciar este capítulo relacionado com a modelação do produto, é importante dar a conhecer as várias metodologias utilizadas atualmente neste contexto, de modo a enquadrar as diferentes secções aqui elaboradas.

Todo o desenvolvimento de software requer uma metodologia, que por norma é como um guia para o mesmo e que também pode ser usado no auxílio da resolução de problemas. Refira-se que esta metodologia, é constituída por fases, tarefas, métodos, técnicas e ferramentas. [37]

Um projeto de Engenharia de software, por norma, na sua forma mais geral, é constituído pelas seguintes fases:[7]

- **Iniciação** - estudo com o objetivo de perceber se o projeto é viável ou não, garantindo as condições para o seu sucesso;
- **Análise** - recolha de requisitos do sistema em função das necessidades existentes;
- **Estruturação** - definição de cada módulo do programa, bem como da segurança e das transações com a base de dados, e ainda de como o sistema irá funcionar com o utilizador;

- **Implementação** - criação do software, com a escrita do código em si e construção da base de dados com a povoação da mesma;
- **Validação** - teste do sistema, verificando se o sistema cumpre os requisitos dos utilizadores e se funciona como pretendido;
- **Revisão e Evolução** - nesta fase o objetivo passa por perceber se existem erros consideráveis, e se é necessário corrigir de acordo com as especificações pedidas. Envolve também, se necessário, a adição de novas tarefas e funções de forma a aumentar as funcionalidades do sistema.

Existem vários modelos utilizados pela engenharia de software onde se destacam os seguintes<sup>1</sup>:

**Waterfall:** Este define o conjunto de passos que devem ser seguidos na construção de um sistema, os quais devem ocorrer por uma ordem pré-definida, com um ponto de situação no fim de cada etapa e imediatamente antes de uma outra ser iniciada. É uma abordagem de desenvolvimento sequencial através de várias fases, onde cada fase recebe informações da anterior e esta, por sua vez, fornece à seguinte. O propósito deste modelo de desenvolvimento de sistemas é a divisão do processo em partes que podem ser geridas e que relacionadas entre si, criam uma estrutura organizada. É, portanto, uma representação do que acontece durante o desenvolvimento do projeto, que proporciona uma base introdutória ao desenvolvimento de um sistema, desde que se verifique ocorrência de todas as fases identificadas no modelo [7].

**Incremental:** É visto como uma evolução do modelo *Waterfall*, onde este último vai sendo incrementado em cada fase. O projeto é dividido em partes mais simples e pequenas e é aplicado o princípio do modelo *Waterfall* a cada uma delas. Assim, em cada parte do projeto, assiste-se a uma ligeira modelação, implementação e teste até que o produto esteja finalizado [38].

---

<sup>1</sup>optou-se pelos termos em inglês pela dualidade de conceitos obtidos na tradução. Desta forma mantém-se o verdadeiro significado do termo.

**Prototyping:** Modelo que se baseia na criação de protótipos de software, ou seja, versões incompletas de software que vão sofrendo evoluções. O software atravessa diferentes versões que vão sofrendo avaliações até chegar a uma versão final que satisfaça as necessidades inicialmente existentes [39].

**Agile:** É baseado num desenvolvimento iterativo e incremental, que tem como principais características promover o trabalho em equipa, colaboração, e adaptação do projeto de acordo com a evolução do mesmo. Segue os princípios de comunicação cara-a-cara com o cliente, incluindo uma troca assídua de resultados de modo a receber constantes *feedbacks*. Desta forma, é um método que tem como princípio a divisão das tarefas em pequenos incrementos com o menor planeamento possível, não incluindo assim um planeamento a longo prazo[40].

Apesar de ser um método de grande importância e com grande eficiência atualmente, não é aquele que se encaixa melhor neste projeto, pois o protótipo desenvolvido é para apresentação e demonstração de uma tecnologia, e não para servir os interesses de um determinado cliente final. De qualquer forma, é um método a seguir no caso de, futuramente, se alterar esta plataforma de acordo com as necessidades de um cliente alvo.

Por norma, o cumprimento destes passos e a elaboração de qualquer tipo de planificação, é pedida e exigida a profissionais que trabalham em equipa na elaboração de grandes projetos de software. Em casos de programação a um nível não tão elevado, como é o caso, não se exige o cumprimento destes passos, mas se é pretendida alguma organização e eficiência, é sempre positivo que isso aconteça, em prol de um trabalho melhor. Assim, tendo em atenção as descrições dos modelos acima apresentados e os passos efetuados para a elaboração deste projeto, verifica-se uma ligação à filosofia do modelo **Waterfall** e do **Prototyping**.

## 4.2 Utilizadores

Esta aplicação está projetada para se adaptar apenas a um tipo de utilizador e, como tal, apenas um tipo de cenário foi pensado. Quem gere as máquinas de um determinado hospital, por norma, é o corpo técnico. Por conseguinte, faz sentido que apenas estes façam uso da plataforma de monitorização, pois em caso de anomalia são eles os contactados. A aplicação foi, portanto, pensada para um tipo de utilizador que tenha grande interação com componentes informáticos e que possua uma vasta experiência a nível de comunicação na internet, com grande capacidade de aprender os conceitos da plataforma, de modo a interpretar com facilidade as notificações produzidas pela aplicação, e consequentemente resolver com brevidade os problemas.

## 4.3 Casos de Uso

Como já mencionado anteriormente, o objectivo foi desenvolver uma aplicação de monitorização de sistemas constituída por três componentes essenciais, Unidade de Análise, Unidade de Processamento, e Unidade de Visualização que tem as seguintes funcionalidades gerais:

- Gerar alertas para os utilizadores;
- Ver máquinas e respetivas informações e desempenhos;
- Visualizar notificações geradas;
- Agendar notificações.

As funcionalidades acima referidas generalizam a totalidade das funcionalidades que existem na aplicação, isto é, todas as restantes a identificar encaixam-se numa destas funcionalidades genéricas.

A cada uma das funcionalidades genéricas enumeradas acima, corresponde um diagrama de casos de uso onde são identificadas as funcionalidades individuais do sistema que encaixam na funcionalidade genérica em causa.

O diagrama de Casos de Uso apresentado na Figura 4.1 diz respeito ao diagrama que resume a aplicação na sua forma global. É de notar que este

representa as funcionalidades genéricas enunciadas no início desta secção, as quais também possuem o seu diagrama individual. Esse diagrama comporta as sub-funcionalidades que se encaixam na funcionalidade genérica então descrita.

Os casos de uso individuais, já referidos no diagrama de casos de uso da plataforma, estão aqui identificados devido à importância que representam e ao seu relacionamento com os restantes diagramas de caso de uso. É de salientar que, sem que algumas destas funcionalidades sejam acionadas, não será possível aceder às sub-funções que estão presentes nos restantes diagramas de casos de uso. Por exemplo, se o utilizador não se registar, ou mesmo se não estiver autenticado na plataforma, não terá possibilidade de efetuar qualquer tipo de interação com a aplicação.

Os casos de uso individuais identificados no diagrama de casos de uso da plataforma são de fácil compreensão, uma vez que o nome não deixa grandes dúvidas acerca da funcionalidade que esquematiza. No entanto, apresenta-se uma curta descrição para cada um dos que são mais relevantes para o sistema.

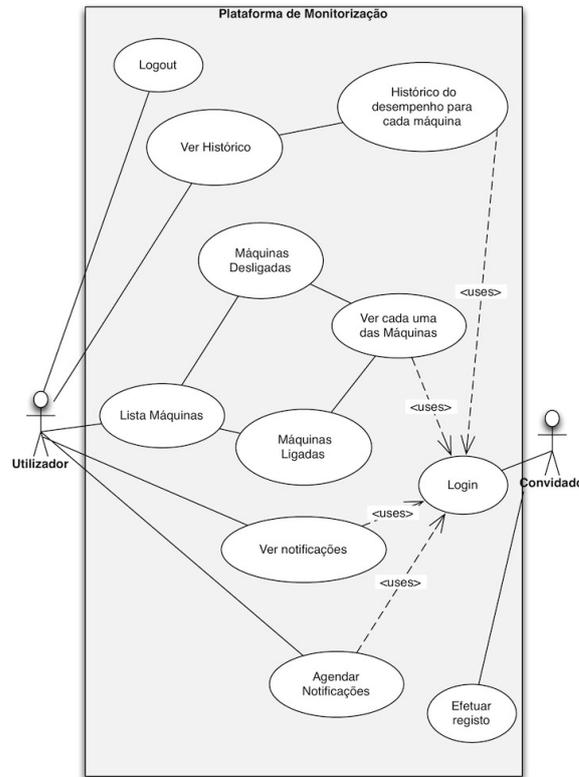


Figura 4.1: Diagrama Casos de Uso da aplicação de Monitorização

### 4.3.1 Caso de Uso Efetuar Registo

O caso de uso Efetuar Registo é um dos casos de uso do diagrama apresentado em cima que é considerado crucial, pois representa a funcionalidade que permite aos utilizadores registarem-se na plataforma. É, portanto, neste caso de uso que definimos quais as informações que o utilizador terá que fornecer para se tornar utilizador. Por outro lado, este é o primeiro verdadeiro contacto do utilizador com a plataforma, o que por si só já é considerado importante.

Para o registo na plataforma são exigidos três elementos de informação ao novo utilizador: o endereço de email, a palavra-passe de acesso e um nome de utilizador. Os campos que dizem respeito ao nome de utilizador serão para dialogar com o utilizador de forma mais clara. Por sua vez, o nome de

utilizador e a palavra-passe de acesso serão utilizados para poder ser feita uma autenticação no sistema. O endereço de email tem ainda interesse para comunicar com o utilizador, por exemplo, para lhe enviar os alertas acerca de notificações agendadas ou fazer a validação do próprio registo.

Tabela 4.1: Fluxo de eventos relativos ao Caso de Uso Efetuar Registo

| <b>Caso de Uso</b>                                  | <b>Registrar</b>  |                                  |   |
|---|---|----------------------------------|---|
| <b>Descrição</b>                                    | Novo utilizador pretende usar plataforma de monitorização |                                  |   |
| <b>Condições</b>                                    | Deverá possuir um email                                   |                                  |   |
| <b>Fluxo de eventos</b>                             |   | <b>Utilizador</b>                | <b>Sistema</b>  |
|   | 1   | Utilizador faz pedido de registo |   |
|   | 2   |                                  | Sistema apresenta formulário de registo                     |
|   | 3   |                                  | Solicita nome de utilizador                                 |
|   | 4   | Introduz nome de utilizador      |   |
|   | 5   |                                  | Solicita introdução de email                                |
|   | 6   | Introduz email                   |   |
|   | 7   |                                  | Valida introdução de email                                  |
|   | 8   |                                  | Solicita Password   |
|   | 9   | Introduz password                |   |
|   | 10  |                                  | Solicita reintrodução de password                           |
|   | 11  | Inserir password                 |   |
|   | 12  |                                  | Valida password   |
|   | 13  |                                  | Valida formulário   |
| 14  |   | Regista dados                    |   |
|   |   | <b>Utilizador</b>                | <b>Sistema</b>  |
| <b>Alternativa: email mal formulado</b>             | 1   |                                  | Avisa que endereço de email está mal formado                |
|   | 2   |                                  | Limpa campos e volta ao passo 6                             |
|   |   | <b>Utilizador</b>                | <b>Sistema</b>  |
| <b>Alternativa: password não alfanumérica</b>       | 1   |                                  | Avisa que password deverá ser alfanumérica e com caracteres |
|   | 2   |                                  | Limpa campos de password e volta ao passo 9                 |
|   |   | <b>Utilizador</b>                | <b>Sistema</b>  |
| <b>Alternativa: campo utilizador não preenchido</b> | 1   |                                  | Avisa que nome de utilizador está por preencher             |
|   | 2   |                                  | Limpa campos  |
|   | 3   |                                  | Volta ao passo 3  |

### 4.3.2 Caso de Uso Login

A partir do momento em que um utilizador passa a estar registado está apto a usufruir de todas as funcionalidades da plataforma. No entanto, para tal, precisa de se autenticar, de modo a ter acesso e para que o sistema possa oferecer as funcionalidades ao utilizador em causa. Para o utilizador se autenticar é necessário que faculte uma combinação de endereço de email

e password válidas para o sistema. Caso este se esqueça da sua password, pode sempre ativar o caso de uso Recuperar Password.

Tabela 4.2: Fluxo de eventos relativos ao Caso de Uso Login

| Caso de Uso                         | <u>Login</u>   |                            |  |
|-------------------------------------|--|----------------------------|--|
| Descrição                           | Utilizador pretende autenticar-se para poder aceder as varias funcionalidades do sistema |                            |  |
| Condições                           | Utilizador já se deve encontrar registado no sistema                                     |                            |  |
| Fluxo de Eventos                    |  | Utilizador                 | Sistema  |
|                                     | 1  | Pretende autenticar-se     |  |
|                                     | 2  |                            | Solicita endereço e-mail   |
|                                     | 3  | Introduz endereço de email |  |
|                                     | 4  |                            | Solicita password  |
|                                     | 5  | Introduz password          |  |
|                                     | 6  | Solicita autenticação      |  |
|                                     | 7  |                            | Valida autenticação  |
|                                     | 8  |                            | Regista autenticação   |
|                                     | 9  |                            | Carrega <u>Home Page</u> da plataforma com as informações daquele utilizador |
| Alternativa:<br>Autenticação Errada |  | Utilizador                 | Sistema  |
|                                     | 1  |                            | Limpa campos de <u>Login</u>   |
|                                     | 2  |                            | Aviso sobre a invalidade dos campos  |
|                                     | 3  |                            | Volta ao passo 2 do procedimento normal                                      |

### 4.3.3 Caso de Uso Logout

Depois de se autenticar, o utilizador pode querer abandonar o sistema, pelo que pode comunica-lo ao sistema acionando o caso de uso Logout.

Tabela 4.3: Fluxo de eventos relativos ao Caso de Uso Logout

| Caso de Uso      | <u>Logout</u>  |  |                                  |
|------------------|--|--|----------------------------------|
| Descrição        | Utilizador pretende abandonar o sistema e fechar o acesso á sua informação |  |                                  |
| Condições        | Utilizador já se deve encontrar autenticado no sistema                     |  |                                  |
| Fluxo de Eventos |  | Utilizador                               | Sistema                          |
|                  | 1  | Informa que pretende fazer <u>logout</u> |                                  |
|                  | 2  |  | Identifica Utilizador            |
|                  | 3  |  | Regista <u>Logout</u>            |
|                  | 4  |  | Apresenta página de <u>Login</u> |

#### 4.3.4 Diagrama Caso de Uso Ver Máquina

Tendo em conta que a funcionalidade Ver Máquina é uma das principais fontes de informação da plataforma, considera-se este caso de uso um dos mais importantes do sistema, como já foi referido, tendo assim particular atenção.

Como se pode verificar pelas relações existentes no diagrama apresentado na Figura 4.2, depois de escolher a máquina pretendida, será possível selecionar o tipo de dados que o utilizador pretende ver, podendo optar por dados de hardware, dados de processos e dados de rede. Ao escolher cada uma das opções será então possível visualizar distintos tipos de informação relativamente aos recursos utilizados por essa mesma máquina.

Se optar por visualizar os dados sobre os Processos será apresentada uma lista daqueles que estão ativos, bem como os recursos, memória e processador, que cada um está a utilizar naquele exato momento. Caso opte pela opção Hardware terá uma versão detalhada do desempenho em tempo real dos componentes. Em relação aos Dados de Rede poderá visualizar as trocas de dados que estão a ocorrer entre a máquina e o exterior, via Internet.

Uma outra função oferecida será o acesso a dados históricos relativamente a cada uma das máquinas. Haverá possibilidade de consultar os dados relativos ao desempenho relativamente a um período de tempo definido pelo utilizador.

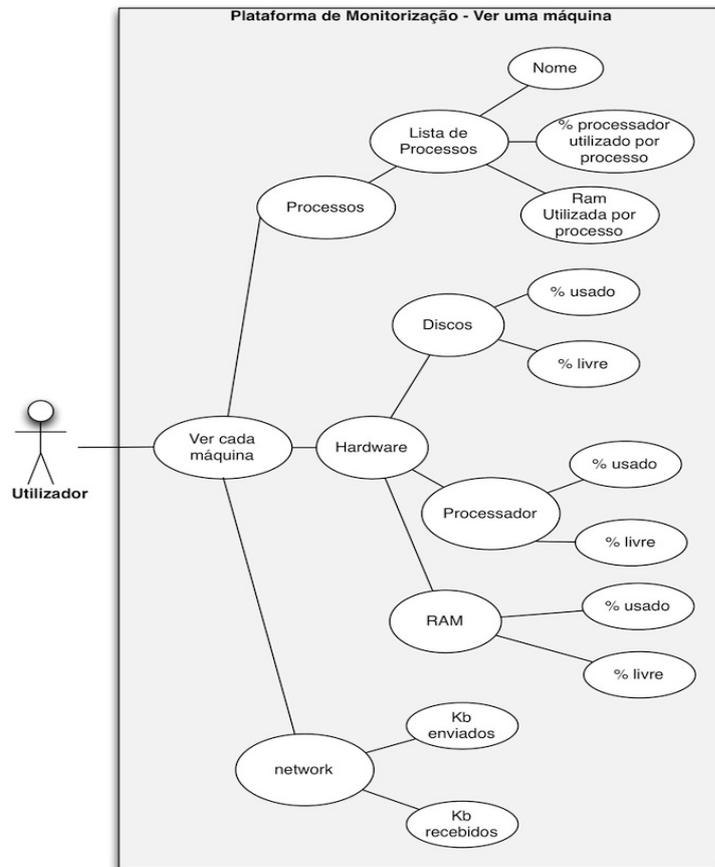


Figura 4.2: Diagrama Case de Uso Ver Máquina

### 4.3.5 Diagrama Caso de Uso Notificações

Neste caso, observando o diagrama para o Caso de Uso Notificações (Figura 4.3), o utilizador poderá optar por ver as notificações já geradas, agendar ser notificado para uma outra situação ou remover determinada notificação que estava anteriormente agendada. Na opção Ver Notificações, poderá ver uma lista das notificações, os relatórios detalhados relativos ao tipo de falha verificado e ainda o utilizador que recebeu o alerta.

Para adicionar uma nova notificação, o utilizador precisa de facultar ao sistema o endereço de email para o qual pretende receber o alerta, seleccionar o tipo de notificação que vai receber (exemplo: se é um processo), bem como

outros parâmetros do sistema que serão necessários para produzir um alerta. Este tipo de parâmetros refere-se, por exemplo, ao tempo de execução de um determinado processo, pois o sistema tem capacidade de monitorizar processos temporários. É necessário, portanto, que o utilizador introduza o tempo de execução, porque, se o tempo pré-definido terminar e o processo ainda continuar em execução, será gerado um alerta. Em caso contrário, para remover uma notificação, será apenas necessário seleccionar a mesma da lista de notificações agendadas, e submeter a ação.

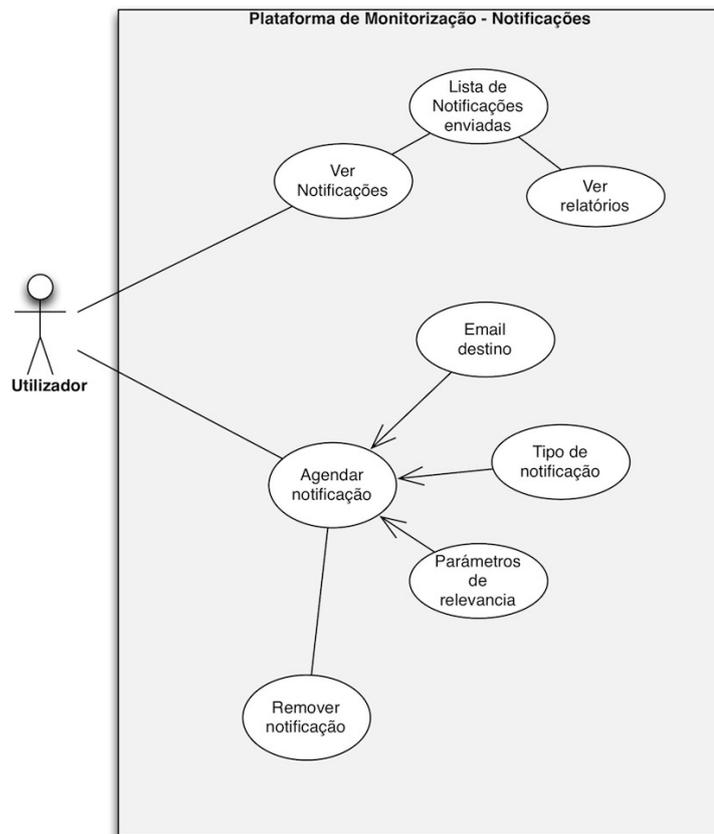


Figura 4.3: Caso de Uso Notificações

## 4.4 Requisitos

Nesta secção serão apresentados os requisitos funcionais e não funcionais que se pretende implementar na plataforma de monitorização, aquando do desenvolvimento da mesma.

### 4.4.1 Requisitos Funcionais

Assim, os requisitos funcionais estão apresentados nas tabelas 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12, 4.13, 4.14, 4.15, 4.16, 4.17, 4.18, 4.19, 4.20.

Tabela 4.4: Requisitos Funcionais: Identificador 1

|            | Identificador 1  |
|------------|--|
| DESCRIÇÃO  | O sistema terá que ser capaz de notificar o utilizador em caso de falha e erro nas máquinas monitorizadas. |
| RAZÃO      | Um dos objectivos do sistema é a notificar o utilizador e alertar o mesmo de possíveis erros e falhas.     |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.5: Requisitos Funcionais: Identificador 2

|            | Identificador 2  |
|------------|--|
| DESCRIÇÃO  | O Utilizador deve poder consultar, pesquisar e explorar todas as máquinas monitorizadas naquele momento. |
| RAZÃO      | Um dos objectivos chave do sistema é a disponibilização de informação por forma visual.                  |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.6: Requisitos Funcionais: Identificador 3

|            | Identificador 3  |
|------------|--|
| DESCRIÇÃO  | O Utilizador deve poder consultar o histórico sobre as máquinas já monitorizadas anteriormente.  |
| RAZÃO      | Nenhuma ação deve ser totalmente apagada. A razão para a mesma passa por utilizar informações anteriores para se poderem tomar novas decisões. |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.7: Requisitos Funcionais: Identificador 4

|            | Identificador 4  |
|------------|--|
| DESCRIÇÃO  | O Utilizador deve poder consultar os parâmetros de sistema atuais das máquinas, tais como hardware, processos executados na máquina, e conexões de rede. |
| RAZÃO      | Um dos objectivos chave do sistema é a disponibilização de informação por forma visual.  |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.8: Requisitos Funcionais: Identificador 5

|            | Identificador 5  |
|------------|--|
| DESCRIÇÃO  | O Utilizador deve poder consultar o histórico dos parâmetros de sistema das máquinas, tais como hardware, processos executados na máquina, e conexões de rede. |
| RAZÃO      | Um dos objectivos chave do sistema é a disponibilização de informação anterior para futura utilização.   |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.9: Requisitos Funcionais: Identificador 6

|            | Identificador 6  |
|------------|--|
| DESCRIÇÃO  | Sistema deve permitir ao utilizador introduzir o intervalo de tempo (datas) para as quais pretende ver o histórico das máquinas. |
| RAZÃO      | Um dos objectivos chave do sistema é a facilidade na procura de informação anterior.   |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.10: Requisitos Funcionais: Identificador 7

|            | Identificador 7  |
|------------|--|
| DESCRIÇÃO  | Utilizador pode consultar detalhes sobre os dados de sistema (atuais ou histórico) sob a forma de gráficos ilustrativos.                     |
| RAZÃO      | Um dos objectivos chave do sistema é a disponibilização de informação por forma visual, de fácil interpretação e atrativo para o utilizador. |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.11: Requisitos Funcionais: Identificador 8

|            | Identificador 8  |
|------------|--|
| DESCRIÇÃO  | O Utilizador deve poder agendar determinadas notificações.   |
| RAZÃO      | Um dos objectivos do sistema é permitir agenda notificações. |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.12: Requisitos Funcionais: Identificador 9

|            | Identificador 9  |
|------------|--|
| DESCRIÇÃO  | Sistema deve permitir ao utilizador escolher ser notificado devido uma determinada falha mais especifica, como falha em processos de ordem temporária. |
| RAZÃO      | O Utilizador deverá poder tomar decisões conforme a necessidade dos acontecimentos.  |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.13: Requisitos Funcionais: Identificador 10

|            | Identificador 10  |
|------------|---|
| DESCRIÇÃO  | Sistema deverá ter a capacidade de enviar alertas para o utilizador sob a forma de email. |
| RAZÃO      | Comunicação entre o sistema e o utilizador.   |
| ORIGEM     | Stakeholder   |
| PRIORIDADE | 5   |

Tabela 4.14: Requisitos Funcionais: Identificador 11

|            | Identificador 11   |
|------------|--|
| DESCRIÇÃO  | Ao agendar uma nova notificação, o sistema deve perguntar ao Utilizador qual é a máquina para que se pretende a notificação, bem como o email para onde enviar o tal alerta. |
| RAZÃO      | Para enviar um alerta sobre determinada falha, sistema tem que saber a identificação da máquina bem como o email do utilizador.  |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.15: Requisitos Funcionais: Identificador 12

|            | Identificador 12   |
|------------|--|
| DESCRIÇÃO  | Sistema deverá ser capaz de gerar relatórios de estado com os detalhes da máquina aquando da falha verificada. |
| RAZÃO      | É importante disponibilizar ao utilizador informação detalhada sobre um determinado problema.                  |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.16: Requisitos Funcionais: Identificador 13

|            | Identificador 13   |
|------------|--|
| DESCRIÇÃO  | Utilizador deverá poder consultar uma lista de notificações já produzidas e enviadas, bem como os relatórios associados.                                     |
| RAZÃO      | É importante armazenar informação detalhada sobre um determinado problema já ocorrido anteriormente para que o utilizador possa consultar quando necessitar. |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.17: Requisitos Funcionais: Identificador 14

|            | Identificador 14   |
|------------|--|
| DESCRIÇÃO  | Utilizador deverá poder cancelar alertas já agendados caso não pretenda ser mais notificado sobre determinado assunto. |
| RAZÃO      | O Utilizador deve ter a possibilidade de gerir, da forma que entender, o seu sistema de dados.                         |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.18: Requisitos Funcionais: Identificador 15

|            | Identificador 15  |
|------------|---|
| DESCRIÇÃO  | Sistema deverá permitir ao utilizador saber se uma determinada máquina se encontra ligada ou desligada. |
| RAZÃO      | Importante o utilizador saber o estado das máquinas.  |
| ORIGEM     | Stakeholder   |
| PRIORIDADE | 5   |

Tabela 4.19: Requisitos Funcionais: Identificador 16

|            | Identificador 16   |
|------------|--|
| DESCRIÇÃO  | Utilizador deverá poder eliminar notificações que já foram geradas e enviadas, bem como os relatórios. |
| RAZÃO      | O Utilizador deve ter a possibilidade de gerir, da forma que entender, o seu sistema de dados.         |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | 5  |

Tabela 4.20: Requisitos Funcionais: Identificador 17

|            | Identificador 17  |
|------------|---|
| DESCRIÇÃO  | Sistema deverá ter capacidade de avisar o utilizador em caso de resolução de uma situação anómala, pela qual tenha sido notificado. |
| RAZÃO      | Sistema terá que ser inteligente e autónomo.  |
| ORIGEM     | Stakeholder   |
| PRIORIDADE | 5   |

### 4.4.2 Requisitos Não Funcionais

Os requisitos não funcionais que serão desenvolvidos na plataforma de monitorização encontram-se descritos nas tabelas 4.21, 4.22, 4.23.

Tabela 4.21: Requisitos Não Funcionais: Identificador 18

|            | Identificador 18   |
|------------|--|
| DESCRIÇÃO  | O sistema de base de dados deve ser desenvolvido para a tecnologia Microsoft SQL Server. |
| RAZÃO      | Especificidades do Servidor.   |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | -  |

Tabela 4.22: Requisitos Não Funcionais: Identificador 19

|            | Identificador 19   |
|------------|--|
| DESCRIÇÃO  | O sistema de base de dados deve ser centralizado e disponível para ser acedido por redes externas. |
| RAZÃO      | Globalização do Sistema  |
| ORIGEM     | Stakeholder  |
| PRIORIDADE | -  |

Tabela 4.23: Requisitos Não Funcionais: Identificador 20

|            | Identificador 20  |
|------------|---|
| DESCRIÇÃO  | O sistema deve proporcionar uma experiência agradável e deve ser de fácil utilização. |
| RAZÃO      | Look & Feel.  |
| ORIGEM     | Stakeholder   |
| PRIORIDADE | -   |

## 4.5 Síntese

Esta capítulo resumiu todos os dados necessários e pensados na fase modelação, clarificando-se assim o que se iria desenvolver. Verifica-se que a plataforma tem um conjunto de requisitos, cada um com um grau de importância associado, que permitirá implementar estes mesmo requisito pela sua ordem importância. Deste modo, criou-se um guia para se seguir durante as várias fases da implementação.



# Capítulo 5

## Desenvolvimento do Produto

### 5.1 Linguagens Utilizadas

Na implementação das várias estruturas constituíntes desta plataforma de monitorização foram utilizadas várias linguagens, sendo o C# a base de toda a aplicação. O C# é uma linguagem simples, moderna e orientada aos objetos, com a sua origem na família de linguagens C, C++ e é também muito familiar para os programadores em Java [41]. A escolha recaiu sobre esta linguagem pela sua crescente utilização no desenvolvimento dos mais variados aplicativos informáticos, o que favorece a interação, interoperabilidade e compatibilidade da aplicação com a maioria das máquinas e computadores utilizados nos dias de hoje. Ainda assim, é uma linguagem atual e com grande margem de progressão num futuro próximo, quer pela facilidade de desenvolvimento que apresenta, quer pelo seu carácter intuitivo para com o utilizador [42]. Outra grande vantagem desta linguagem, foi o seu desenvolvimento específico para .NET, o que permitiu um alargado uso por parte dos programadores na construção de plataformas direcionadas para uso web [42]. A linguagem C# foi escolhida também pelos recursos que esta linguagem oferece para criar aplicações para Windows, uma vez que a estrutura desenvolvida tem como finalidade ser inserida num Hospital no Norte do País, que

tem o Windows como sistema operativo das principais máquinas.

Esta linguagem foi aplicada na programação de todo o código de suporte que oferece funcionalidade à aplicação. Para além do C#, foram ainda utilizadas linguagens como o HTML (plataforma ASP.NET), PHP e SQL.

No que respeita à construção da parte visual e de interação com o utilizador a escolha recaiu sobre o ASP.NET, pela sua ligação e combinação com o C#. Em combinação com o C#, ASP.NET é na atualidade a tecnologia mais popular para o desenvolvimento de *websites* [43]. ASP.NET é uma plataforma com uma estrutura de bibliotecas que permite, através de uma linguagem de programação integrada, a criação de páginas dinâmicas, bem como uma interatividade completa entre *websites* e bases de dados num servidor, contribuindo para uma melhoria nas transações online [43]. ASP.NET tem assistido a uma grande afirmação no mundo *web* combinando aparência e um design atrativo com funcionalidade. A escolha do ASP.NET traz ainda, como vantagens, compatibilidade com uma diversidade de linguagens como HTML, JScript e PHP.

Recorreu-se também ao uso de AJAX, que é uma tecnologia usada no lado Cliente da aplicação, que visa proporcionar comunicações assíncronas entre o utilizador e o servidor. Deste modo, torna possível o envio e retorno de informação em ambiente *background* sem interferir com o que utilizador esta a ver no momento, ou seja, não necessita de se fazer *refresh* à página *web* para mostrar novo conteúdo. Em modo de sumário, para que isto seja possível, AJAX incorpora:[44]

- Apresentação fazendo uso de XHTML e CSS
- Apresentação e interação de informação dinamicamente fazendo uso do **Document Object Model**<sup>1</sup>
- Manipulação e troca de dados através de XML e XSLT;

---

<sup>1</sup>especificação da W3C, independente de plataforma e linguagem, onde pode-se dinamicamente alterar e editar a estrutura, conteúdo e estilo de um documento eletrónico, permitindo que o documento seja mais tarde processado e os resultados desse processamento, incorporados de volta no próprio documento[45]

- Apresentação de dados de forma assíncrona através **XMLHttpRequest**<sup>2</sup>.
- Ligação de todas as funções anteriores através de JavaScript

De forma a poder armazenar a extensa quantidade de dados necessários procedeu-se à construção de uma Base de Dados que comunica com o sistema por intermédio de *queries* em linguagem SQL. Para tal, fez-se uso do sistema de gestão de bases de dados (SGBD) MySQL como interface. MySQL é um gestor de bases de dados dotado de uma arquitetura com grande utilidade para um largo espectro de situações. Como todos os outros não consegue ser perfeito, mas oferece flexibilidade suficiente para trabalhar em diferentes ambientes exigentes, como é o caso das aplicações *Web* [46]. Ao mesmo tempo, o MySQL tem compatibilidade com variadas aplicações, *data warehouses*, sistemas redundantes exigentes, bem como capacidade para o processamento de complexas transações online [46]. Os gestores de bases de dados disponibilizam ao utilizador uma linguagem simples para proceder à manutenção da base de dados, por meio de inserções, modificações ou remoções de informação, respondendo também a questões que podem ser colocadas pelo utilizador. É uma linguagem estruturada que facilita a consulta aos dados organizados na base de dados, como é exemplo o SQL [47].

A escolha recaiu sobre esta interface e linguagem devido à sua popularidade, e pelas suas características de portabilidade, compatibilidade com C#, PHP, ASP.NET e facilidade de uso para o utilizador.

## 5.2 Frameworks e APIs Utilizadas

### Visual Studio

Para fazer uso das linguagens acima referidas, o Visual Studio era o ambiente que reunia todos os requisitos fundamentais e, como tal, foi o escolhido para o desenvolvimento da aplicação. Oferece capacidade para desenvolver

---

<sup>2</sup>utilizado para enviar requisições HTTP ou HTTPS diretamente para um servidor web e carregar os dados de resposta do servidor diretamente de volta ao script[45]

aplicações de console, interfaces gráficas, *websites*, aplicações *web* entre outras. Tem ainda compatibilidade e suporte por parte de plataformas como Microsoft Windows, Windows Mobile, Windows CE, .NET Framework and .NET Compact Framework. Suporta também diferentes linguagens de programação, entre as quais se destacam o C#, XML/XSLT, HTML/XHTML, JavaScript e o CSS, sendo todas elas utilizadas na elaboração deste projeto [3].

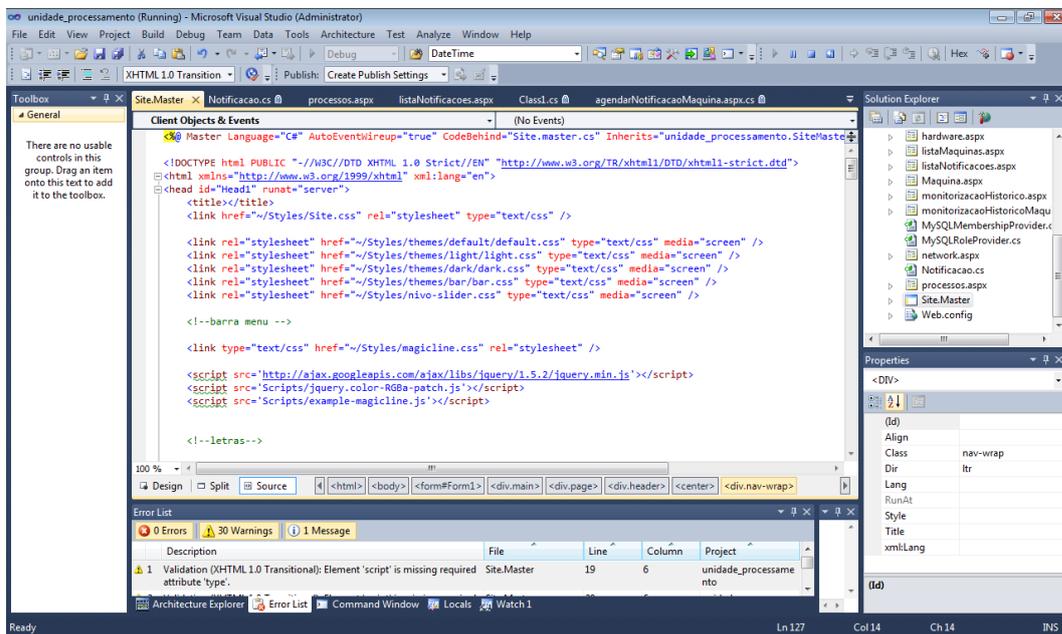


Figura 5.1: Ambiente do Visual Studio durante o desenvolvimento da aplicação em ASP.NET[3]

## Google Chart API

Como o nome indica é uma ferramenta que permite a criação de gráficos a partir de dados e apresentá-os num contexto de uma página *web*. Os gráficos são mostrados sob a forma de classes JavaScript, não sendo necessário qualquer *plugin*. Permite a criação de gráficos altamente interativos para ligação a *dashboards* por exemplo, ou para qualquer outro tipo de função ou integração com uma página *web*. São processados usando tecnologia HTML5 o que permite compatibilidade com qualquer explorador atual. [48]

## WMI

Uma das ferramentas/*framework* com maior importância e contribuição para o desenvolvimento deste trabalho foi o WMI – *Windows Management Instrumentation*, essencialmente utilizado na elaboração da Unidade de Análise, com o intuito de receber dados de um determinado computador.

É uma infra-estrutura baseada nos sistemas operativos Windows, tal como o nome remete, para a monitorização de informação e realização de operações. Permite a escrita de *scripts* e fornece código já estruturado, em distintas linguagens, como Visual Basic e C#, para automatizar tarefas administrativas em computadores. WMI também oferece a possibilidade de gestão de dados noutras partes do sistema operativo, como por exemplo o *System Center Operations Manager*. [4]

É uma ferramenta baseada numa tecnologia padrão o que possibilita aceder a informações nativas num ambiente corporativo. WMI faz uso de *Common Information Model* (CIM), muito conhecido para representar sistemas, aplicações, redes, dispositivos e outros componentes de gestão.

Assim o WMI fornece uma interface que, através de *scripts*, proporciona a obtenção de dados de funcionamento de um sistema, bem como de aplicações locais ou remotas, numa determinada rede. É uma interface desenvolvida de forma uniforme para que não seja necessário fazer uso de uma grande número de APIs para uma determinada operação. [4]

O diagrama da Figura 5.2 mostra as diferentes camadas constituintes da aplicação bem como a relação existente entre as mesmas. Também se verifica os diferentes *providers* e os objetos monitorizados.

Os componentes que constituem o WMI são:[4]

### Objetos monitorizados e *Providers* WMI:

Um *provider* WMI monitoriza um objecto que pode ser um componente físico, como um disco, uma placa de rede, um sistema ou um processo. Assim, um *provider* recebe informação do objecto monitorizado e fornece ao mesmo mensagens provenientes do WMI. Por exemplo, um *provider* existente, em combinação com o diretório *Win32*, que associado a si tem classes tais como *Win32\_Process* ou *Win32\_LogicalDisk*,

forneem informações sobre os processos em curso ou sobre um disco, respetivamente.

**Infra-estrutura WMI:**

É constituída por dois componentes, o *WMI Service* (que inclui o *WMI core*) e o *WMI Repository*. O *WMI Service* funciona como um serviço intermediário entre os *providers*, as aplicações monitorizadas e o *WMI Repository*. Assim, o WMI obtém os dados de forma dinâmica cada vez que é solicitado por uma requisição por parte de um cliente.

**Interface WMI:**

É a aplicação de gestão que interage com a infra-estrutura WMI. Esta aplicação consegue desenvolver *queries*, enumerar dados, executar métodos para os *providers* ou mesmo subscrever eventos chamando APIs.

Deste modo, os *scripts* devolvidos pelo WMI e pelas aplicações de gestão oferecem ao utilizador o mais variado tipo de informação sobre um sistema em diferentes linguagens de programação, como C#, C ou C++ podendo assim fazer-se uso para construção de novas aplicações, como aconteceu para a elaboração da Unidade de Análise .

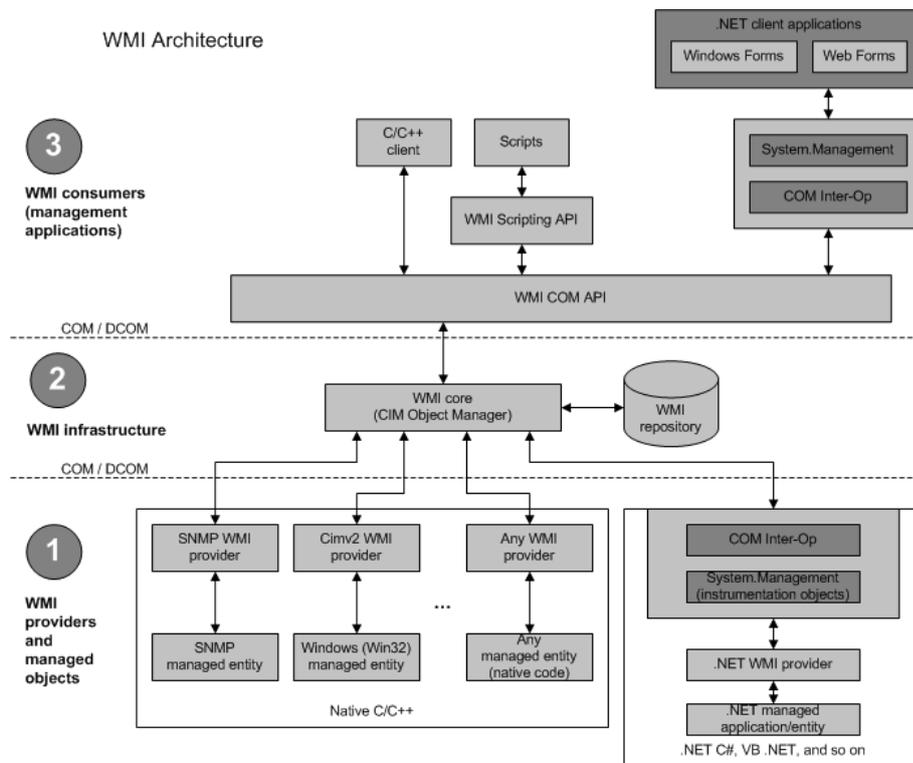


Figura 5.2: Arquitetura WMI - Relação entre a infra-estrutura WMI, WMI *providers* e objetos monitorizados[4]

## ADO.NET

ADO.NET consiste num conjunto de classes que fornece um acesso consistente a bases de dados como o Microsoft SQL Server, bem como a outras bases de dados como OLE DB e ODBC. As aplicações que façam uso de partilha de dados, como é o caso da aplicação desenvolvida, utilizam o ADO.NET para se ligar às bases de dados e recuperar, manipular e atualizar os dados que elas contêm. [5]

ADO.NET permite separar o acesso aos dados da manipulação dos mesmos, podendo estes serem usados separadamente ou em conjunto. ADO.NET inclui provedores de dados .NET FRAMEWORK para se ligar a uma base de dados, de forma a executar comandos e devolver resultados. Estes resultados são diretamente tratados e colocados num *dataset* de modo a ser exposto ao

utilizador. Este *dataset* também pode ser usado independentemente de um provedor .NET FRAMEWORK para lidar com dados locais da aplicação ou mesmo para lidar com um XML que seja usado como base de dados. [5]

ADO.NET fornece aos programadores a possibilidade de desenvolver código da mesma forma e com mesma funcionalidade do modelo nativo, o ADO - ActiveX Data Objects, entre os quais existem algumas diferenças não relevantes para serem explicadas nesta secção [5].

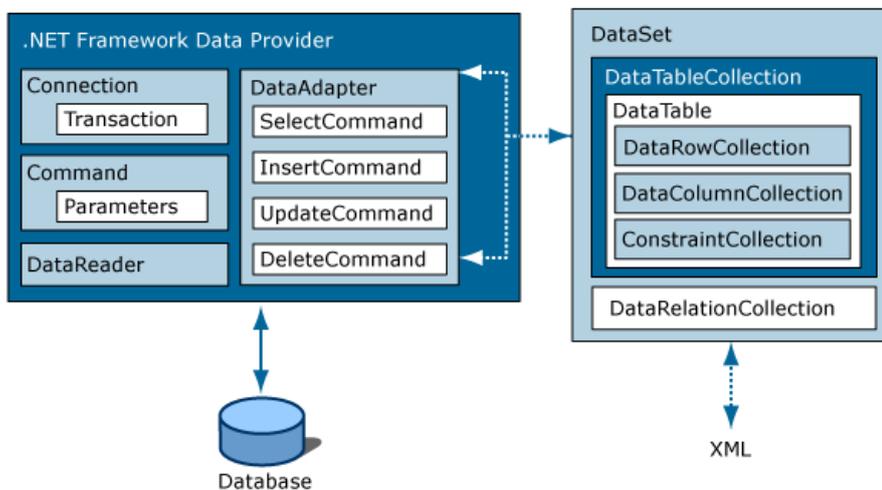


Figura 5.3: Esquema da arquitetura ADO.NET[5]

## 5.3 Componentes Principais

### 5.3.1 Master Page

A *framework ASP.NET*, já abordada e explicada num outro capítulo deste documento, oferece aos seus utilizadores a funcionalidade de Master Pages para um *WebSite*. Master Pages permitem criar uma estrutura consistente na qual se podem inserir diferentes páginas com o seu conteúdo, oferecendo ao *WebSite* uma aparência sofisticada e menos confusa ao orientador aquando da troca de páginas. [49] Fornece também a partilha da linguagem HTML, dos controlos e do código que pode ser usado como *template* para todas as páginas. Assim usando uma Master Page garante-se que todas as páginas

têm elementos comuns, como logotipos, cabeçalhos, rodapés e etiquetas de navegação [49].

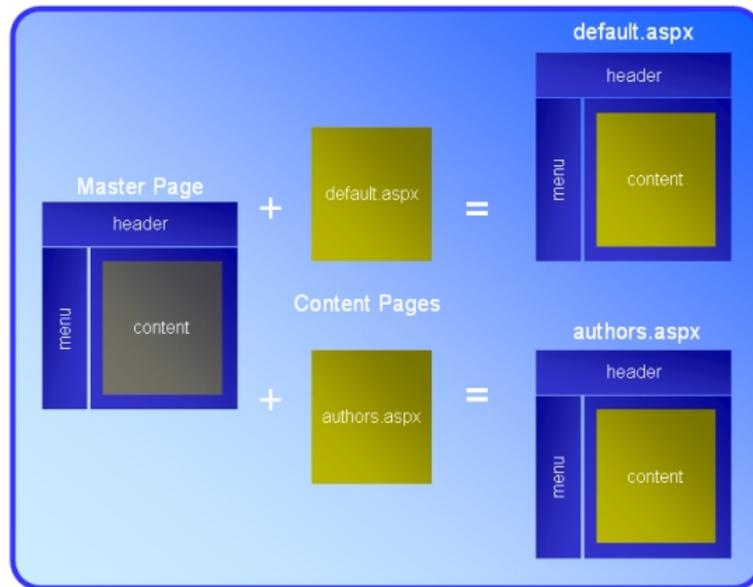


Figura 5.4: Esquema de uma Master Page e seus constituintes[6]

### 5.3.2 Autenticação

Trata-se do processo de certificar que um utilizador tem o estatuto que reclama, que por norma é acompanhado do uso de credenciais ou outra forma de autenticação. O utilizador apresenta as suas credenciais à aplicação ASP.NET, sob a forma de utilizador e password [49].

No caso deste projeto, este conjunto de nomes, passwords, e todas as outras credenciais necessárias está mantido numa Base de Dados SQL com as configurações num ficheiro *web.config*. No ASP.NET, a autenticação é fornecida por intermédio de módulos chamados *authentication providers*. Assim que instalados, estes *providers* estão disponíveis para fazer uso dos ficheiros configuráveis do ASP.NET como o *machine.config* e o *web.config*, acessíveis no diretório raiz da aplicação, permitindo a interação com a base de dados. [49] O *template* da página *web* já tem implementado um formulário de autenticação nativo, sendo apenas necessário acrescentar um *SQL Provider* para

que a autenticação da página fique em comunicação com a Base de Dados.

### 5.3.3 Monitorização em tempo real

Como já foi visto no capítulo 3, esta plataforma permite visualizar o desempenho de cada máquina em tempo real, acompanhando a ação de cada componente de hardware (processador, disco, Ram) à distância, independentemente da localização, apenas precisando de acesso à plataforma através da Internet. Como podemos ver na Figura 5.5, este desempenho é mostrado sob a forma de gráficos, combinando a tecnologia oferecida pela Google Charts API e pelo AJAX que faz com que seja possível visualizar apenas o movimento do gráfico, não sendo necessário constantes *refreshs* à pagina. Permite ainda visualizar percentagens de ocupação de forma mais específica devido aos gráficos estáticos. Neste caso, faz-se uso apenas da API Google Charts, uma vez que não existe necessidade de fazer chamadas assíncronas ao servidor (Figura 5.6).

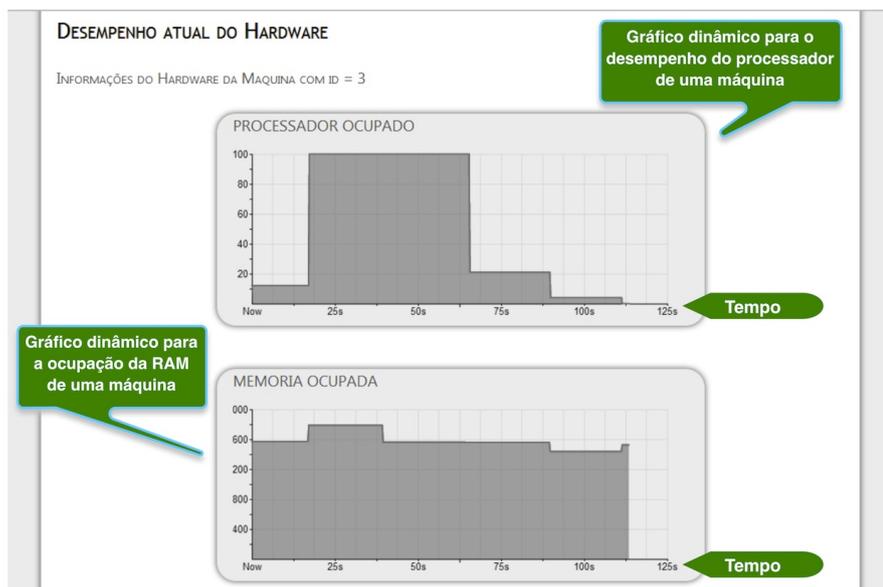


Figura 5.5: Gráficos da monitorização em tempo real, para Processador e RAM

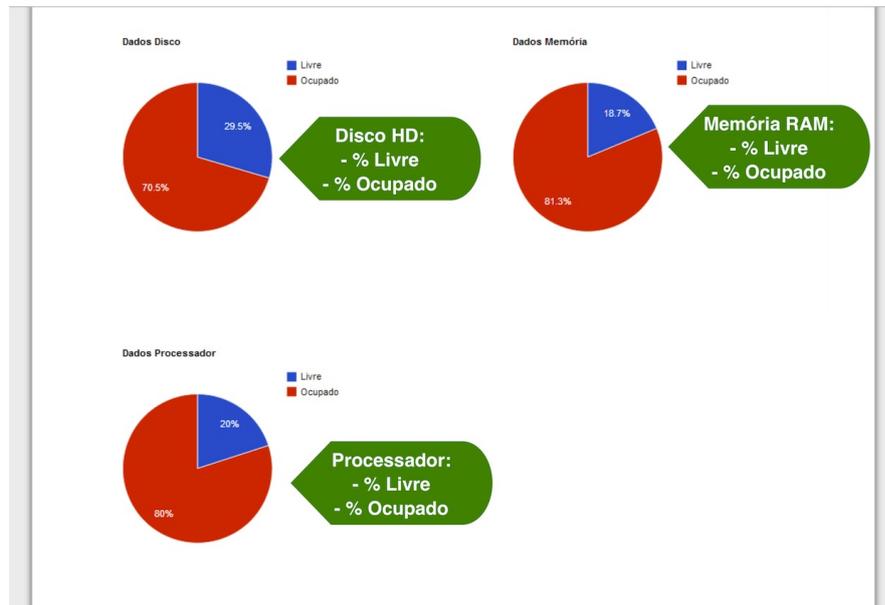


Figura 5.6: Percentagens livre e ocupado em tempo real, para Processador e RAM e Disco

#### 5.3.4 Histórico dos dados monitorizados

Tal como na secção anterior, para a visualização dos dados históricos faz-se uso de gráficos que utilizam a mesma tecnologia descrita acima. A única diferença, como se pode ver na Figura 5.7 está na opção do utilizador poder seleccionar as datas para as quais pretende ver os dados. Depois de o utilizador seleccionar os intervalos para a data, o gráfico apresenta valores mais pormenorizados apenas para esse espaço temporal (Figura 5.8).



Figura 5.7: Gráficos de dados históricos, para Processador e RAM

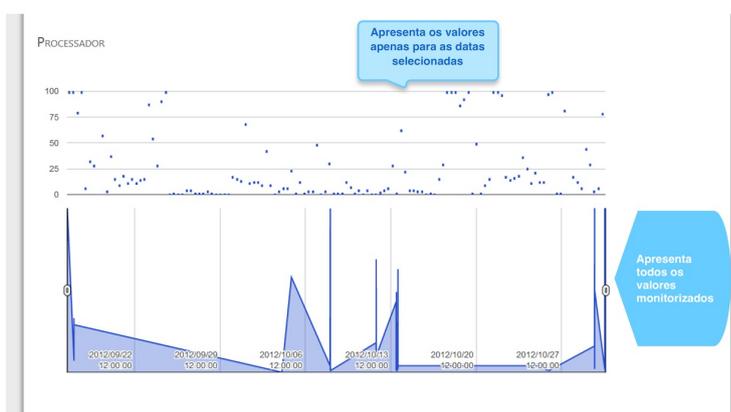


Figura 5.8: Gráficos de dados históricos, para Processador e RAM

### 5.3.5 Agendar Notificação

Como já foi apresentado no capítulo da Visão de Produto, a funcionalidade Agendar Notificação, que permite ao utilizador ser alertado de algum tipo de falha, é uma das grandes vantagens que esta plataforma oferece. Nesta secção, o utilizador deverá escolher, de entre uma lista, um processo para o qual deseje ser notificado, clicando então no botão **Sumbeter** para iniciar a monitorização do mesmo (Figura 5.9). A partir deste momento, a unidade de processamento vigia este processo continuamente. No caso de ser um processo de duração temporária, o utilizador terá de preencher o campo apresentado relativo ao tempo de execução desse processo. Assim, se o processo continuar em execução mais tempo que o introduzido, é enviado um alerta.

Esta funcionalidade faz uso de um utilitário para permitir a restrição de campos, isto é, apenas permite que os campos sejam preenchidos com dados compatíveis. Este controlo é nativo do ASP.NET, como se pode perceber pela figura 5.10, onde é representado um exemplo dessa restrição. As opções que aparecem disponíveis para seleção na *drop down list*<sup>3</sup> são apenas os já existentes ao nível da base de dados.

Dentro da página das notificações, o utilizador também tem a possibilidade de eliminar uma determinada notificação agendada para alerta. Como se verifica na Figura 5.11, apenas é necessário seleccionar a partir de uma lista o processo que se pretende eliminar.

---

<sup>3</sup>elemento de controlo pelo utilizador, o qual permite ao utilizar escolher um valor de uma lista apresentada. Quando inativa apenas apresenta um valor. Quando invocada, apresenta uma lista de valores possíveis



Figura 5.9: Interface para a opção Agendar Notificação

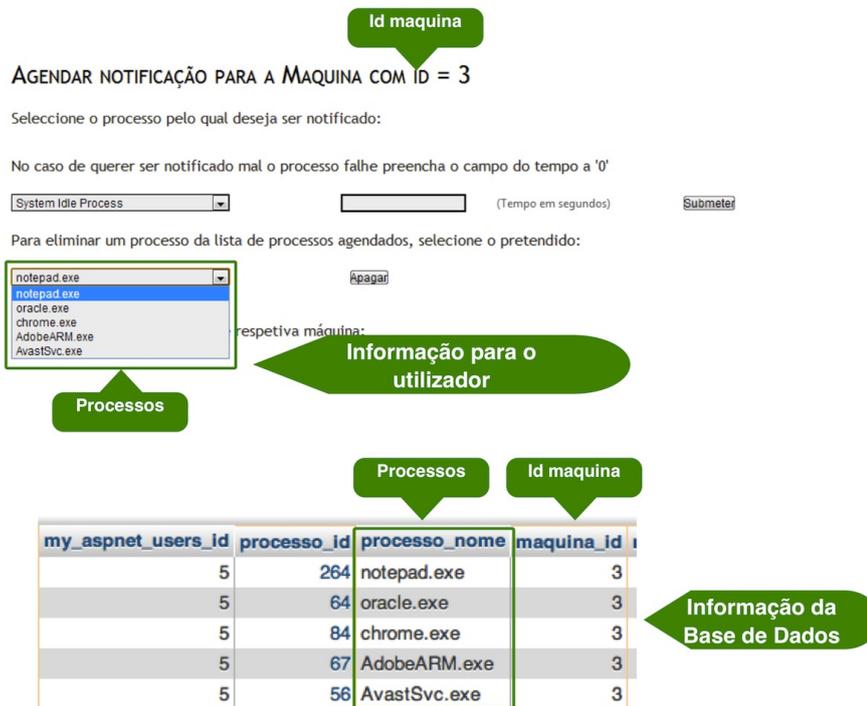


Figura 5.10: Exemplo de *drop down list* constituída pela informação existente na base de dados

HOME MONITORIZAÇÃO ATUAL MONITORIZAÇÃO - HISTÓRICO AGENDAR NOTIFICAÇÃO LISTA DE NOTIFICAÇÕES CONTACTOS

### AGENDAR NOTIFICAÇÃO PARA A MAQUINA COM ID = 3

Selecione o processo pelo qual deseja ser notificado:

No caso de querer ser notificado mal o processo falhe preencha o campo do tempo a '0'

System Idle Process (Tempo em segundos) Submeter

Para eliminar um processo da lista de processos agendados, selecione o pretendido:

notepad.exe  
notepad.exe  
oracle.exe  
chrome.exe  
AdobeARM.exe  
AvastSvc.exe

Apagar

Drop down a usar para eliminar processos já agendados

Lista de alertas agendados

| Nome do Processo | Máquina - MacAddress |
|------------------|----------------------|
| AdobeARM.exe     | 00:0C:29:3D:A7:C6    |
| AvastSvc.exe     | 00:0C:29:3D:A7:C6    |
| chrome.exe       | 00:0C:29:3D:A7:C6    |
| notepad.exe      | 00:0C:29:3D:A7:C6    |
| oracle.exe       | 00:0C:29:3D:A7:C6    |

Showing 1 to 5 of 5 entries First Previous 1 Next Last

Figura 5.11: Campos a utilizar para eliminar uma notificação agendada

### 5.3.6 Envio de Email

Nesta aplicação, onde a monitorização e a necessidade de criar alertas são constantes, era necessário um meio de comunicação simples, acessível e funcional entre a plataforma de monitorização e os seus utilizadores. Para estabelecer essa comunicação, foi implementado um remetente de email na unidade de processamento. Este remetente está sincronizado com esta mesma unidade, para que, quando for desencadeado um novo alerta, seja imediatamente enviada uma mensagem. De acordo com o tipo de erro ou falha existente, é possível notificar apenas um utilizador ou um determinado grupo. Ou seja, no caso de ser uma falha associada a um processo, o email é enviado para o utilizador que agendou esse alerta, caso contrário, se for uma falha associada ao hardware, o alerta é enviado para todos os utilizadores registados na plataforma.

### 5.3.7 Lista de Notificações enviadas

A plataforma permite aos seus utilizadores terem acesso a uma Central de Notificações onde ficam registadas todos alertas enviados, de forma a poderem verificar os erros e as falhas passadas, realizando assim um melhor trabalho, de forma a evitar repetição no futuro. Tal como já foi visto anteriormente esta secção faz uso de tecnologia *Jquery* para atualização automática dos dados de acordo com as novas entradas de informação na base de dados. Pela Figura 5.12, verificamos que é registado uma descrição da falha, a data, a máquina onde a mesma aconteceu e ainda o utilizador que na altura foi alertado.

INFORMAÇÕES DO HISTORIAL DE NOTIFICAÇÕES

| ID | Descrição  | Data                | ID Máquina | Email utilizador         | Tipo Notificação   |
|----|--|---------------------|------------|--------------------------|--|
| 1  | Notificação Processo notepad.exe - Falha Normal para Máquina 3 | 13-10-2012 12:10:38 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Falha Normal para Máquina 3 |
| 2  | Notificação Processo notepad.exe - Falha Normal para Máquina 3 | 13-10-2012 12:18:04 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Falha Normal para Máquina 3 |
| 3  | Notificação Processo notepad.exe - Ressuscitou para Máquina 3  | 13-10-2012 12:20:35 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Ressuscitou para Máquina 3  |
| 4  | Notificação Processo chrome.exe - Falha Normal para Máquina 3  | 13-10-2012 14:02:16 | 3          | luisojeitoso@hotmail.com | Notificação Processo chrome.exe - Falha Normal para Máquina 3  |
| 5  | Notificação Processo notepad.exe - Falha Normal para Máquina 3 | 13-10-2012 14:02:19 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Falha Normal para Máquina 3 |
| 6  | Notificação Processo chrome.exe - Ressuscitou para Máquina 3   | 13-10-2012 14:02:53 | 3          | luisojeitoso@hotmail.com | Notificação Processo chrome.exe - Ressuscitou para Máquina 3   |
| 7  | Notificação Processo notepad.exe - Falha Normal para Máquina 3 | 13-10-2012 14:04:13 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Falha Normal para Máquina 3 |
| 8  | Notificação Processo notepad.exe - Ressuscitou para Máquina 3  | 13-10-2012 14:07:03 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Ressuscitou para Máquina 3  |
| 9  | Notificação Processo notepad.exe - Falha Normal para Máquina 3 | 13-10-2012 14:12:05 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Falha Normal para Máquina 3 |
| 10 | Notificação Processo notepad.exe - Ressuscitou para Máquina 3  | 13-10-2012 14:14:32 | 3          | nunomrvieira@hotmail.com | Notificação Processo notepad.exe - Ressuscitou para Máquina 3  |

Showing 1 to 10 of 10 entries

Developed by NunoVieira -- © Universidade do Minho 2012 - Termos Legais

Figura 5.12: Lista das notificações enviadas para os utilizadores

### 5.3.8 Gerador de Relatórios

Independentemente do contexto em que se está inserido, qualquer tipo de situação anómala detetada exige correção e ainda, mais importante, é perceber a causa dessa mesma situação para prevenção futura. Tendo em conta que o projeto desenvolvido tem como principal função lidar com erros

e falhas nas máquinas, é necessário saber os detalhes de uma anomalia para efetivamente se perceber a causa da mesma. Neste contexto, foi desenvolvido um gerador de relatórios automático. Este é solicitado pela unidade de processamento e cria um documento PDF - Portable Document Format, com os detalhes verificados no momento da falha, sendo posteriormente enviado para o utilizador e também é armazenado na Base de Dados para posterior consulta e estudo de dados.

## 5.4 Interfaces de Utilizador

Apesar da estrutura criada ser constituída por várias unidades, como já foi mostrado e referido em capítulos anteriores, a interface de utilizador resume-se a uma estrutura criada sob a forma de um *Web Site*. Como se trata de uma aplicação desenvolvida especialmente para lidar com a parte técnica das máquinas, a interface de utilizador foi pensada, modelada e desenvolvida para ser usada apenas por um tipo de utilizador, que seriam os responsáveis pela gestão das máquinas. Deste modo, nesta secção, irá ser abordado apenas o único ambiente criado, a Interface de Administrador.

### 5.4.1 Interface Administrador

O utilizador que terá estatuto de administrador terá a possibilidade de aceder e controlar a monitorização das máquinas hospitalares, que são da gestão do mesmo. Depois de fazer *login* terá acesso à *home page* (figura 5.13), onde é oferecida a possibilidade de escolher sobre as tarefas que pretende executar. Pode assim ter acesso à lista de máquinas que estão a ser monitorizadas, bem como à gestão de notificações. Acedendo ao separador lista de máquinas e escolhendo uma delas, tem a possibilidade de verificar qualquer tipo de dados sobre as mesmas. Selecionando a gestão de notificações, pode gerir as notificações já enviadas e os seus relatórios de estado, bem como agendar novas notificações para determinados processos para os quais pretenda ser alertado.

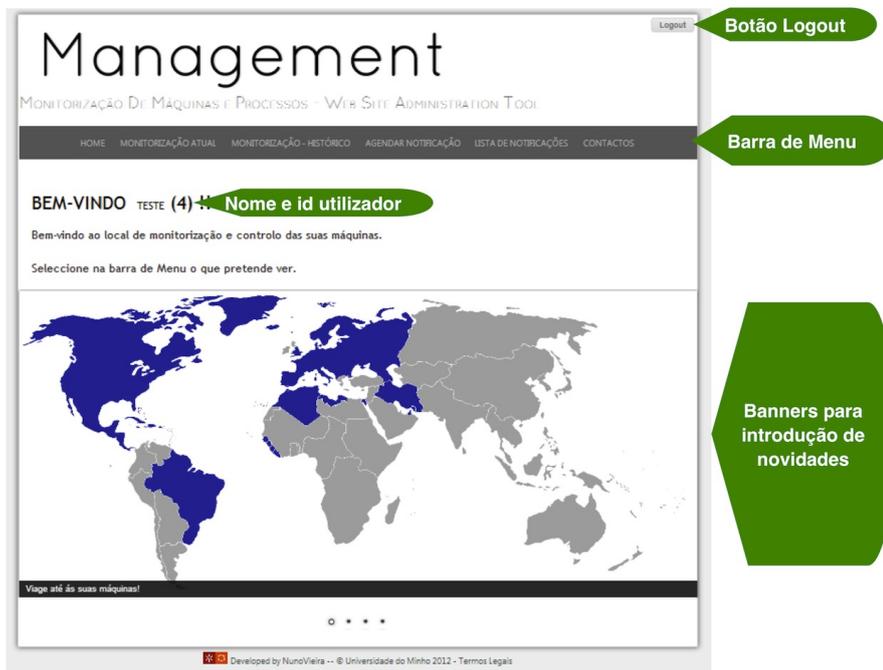


Figura 5.13: Página principal para Administrador

## 5.5 Implementação da Base de Dados

**Definição 1** *Uma base de dados é uma coleção de grande quantidade de dados [47].*

**Definição 2** *Um Sistema Gestor de Bases de Dados (SGBD) é uma aplicação informática desenvolvida para armazenar e gerir bases de dados [47].*

Antes de abordar a temática da implementação da Base de Dados, é importante realçar a importância desta estrutura no contexto do trabalho desenvolvido, quer pelas funcionalidades que oferece, quer pelo fato de ser o grande elo de ligação entre os vários componentes da plataforma criada. Assim, surge a necessidade de criação desta secção onde será explicado e mostrado o modelo relacional construído bem como os fundamentos teóricos que suportam uma estrutura deste tipo.

Em tempos remotos, as bases de dados eram estruturas baseadas em modelos hierárquicos ou modelos em rede. Durante a década de 70, pelo

trabalho de E.F Codd, surgiu um modelo mais simples e flexível com base em relações de dados lógicos, o modelo relacional [50]. A sua implementação foi rápida, contribuindo decisivamente para a massificação da bases de dados nas organizações. O Oracle Server da Oracle Corp., Informix SE e RDS, DB2 da IBM, SQL Server da Microsoft, Sybase SQL da Sybase Inc. entre outros, são exemplos de bases de dados que fazem uso do modelo relacional [47].

O modelo relacional (para esta caso, figura 5.14) trata os dados como relações, representando-os em uma forma de tabelas. Estas são uma estrutura bidimensional, constituída por atributos e tuplos. O esquema concetual de uma relação é o conjunto dos seus atributos, que traduzem o tipo de dados a armazenar [47]. As bases de dados relacionais contêm um ou mais objetos e as tabelas, que armazenam os dados ou informação, sendo esta última transferida por parte das aplicações.

Para se aceder de uma forma mais rápida e fácil à informação contida nas tabelas são definidas chaves, que poderão ser [47]:

**Chave candidata:**

Subconjunto de atributos de uma relação que, em conjunto, identifica univocamente qualquer tuplo, e que não pode ser reduzido sem perder essa qualidade.

**Chave primária:**

Chave selecionada de entre as diversas chaves candidatas, para efetivamente identificar cada tuplo.

**Chave estrangeira:**

Atributo ou conjunto de atributos de uma relação que é chave primária numa outra relação.

Uma entidade corresponde a uma relação ao nível do modelo relacional. Uma entidade será representativa de uma classe de objetos sobre os quais se quer guardar informação [47]. No caso desenvolvido para esta plataforma de monitorização, as entidades são, por exemplo **máquina**, **hardware**, **processo**, etc. Cada instância de uma entidade será caracterizada por um conjunto de atributos. Neste caso concreto, uma máquina tem o seu ID, o seu

macAddr, enquanto um processo tem associado um ID, um Nome e uma Máquina. Entre as entidades estabelecem-se um conjunto de relações que pode ser relação de 1 para 1, de 1 para muitos, ou de muitos para muitos.

Um outro processo importante a ter em conta no desenvolvimento desta base de dados, foi a normalização. O modelo relacional é uma solução que apresenta redundância nos dados, o que provoca perda de espaço, integridade de dados, problemas de manutenção e problemas de desempenho. Assim, para eliminar estas anomalias, recorreu-se ao uso da normalização para garantir uma base de dados eficiente que conseguisse armazenar vários tipos de dados interligados e com velocidade de escrita e leitura.

Resumidamente, a normalização é um conjunto de estados que diminuem a redundância dos dados. Cada estado é denominado de forma normal: [50]

- 1ª Forma Normal: Visa eliminar grupos de valores repetidos para um dado atributo;
- 2ª Forma Normal: Relação em que todos os atributos não pertencentes a qualquer chave candidata devem depender da totalidade da chave;
- 3ª Forma Normal: Relação em que não existem dependências funcionais entre atributos não chave.

É de referir que a escolha da forma normal onde se pretende parar o processo de normalização é uma questão de otimização de inserções, *deletes* e atualizações. Por norma, é a terceira forma normal, mas por vezes pode-se parar numa outra. Depois de cumprir estes procedimentos, consegue-se eliminar muitos erros e reduzir a redundância dos dados. Este foi um exemplo simples de tratamento aplicado ao tipo de dados que existiam, durante a implementação da base de dados.

Além da normalização, durante a modelação desta base de dados fez-se uso de outra técnica, à qual se deu o nome de mapeamento. Devido ao elevado número de tabelas e relações entre ambas, colocou-se em dúvida a eficiência da base de dados aquando de uma chamada à mesma. Assim, construiu-se tabelas denominadas **mapas** apenas para executar e orientar uma comunicação direta a outra tabela. Assim, quando se faz uma *query*

à base de dados, será possível, através das tabelas de mapeamento reunir informações de várias tabelas de forma mais simples e rápida. Como se verifica na Figura 5.14 existe uma tabela de nome **mapa**, que está ligada a outros sub-mapas, como é o caso de **mapa\_hardware**, **mapa\_processo** e **mapa\_rede** que por sua vez estão relacionados com a tabela **hardware**, **processo** e **rede**, respetivamente.

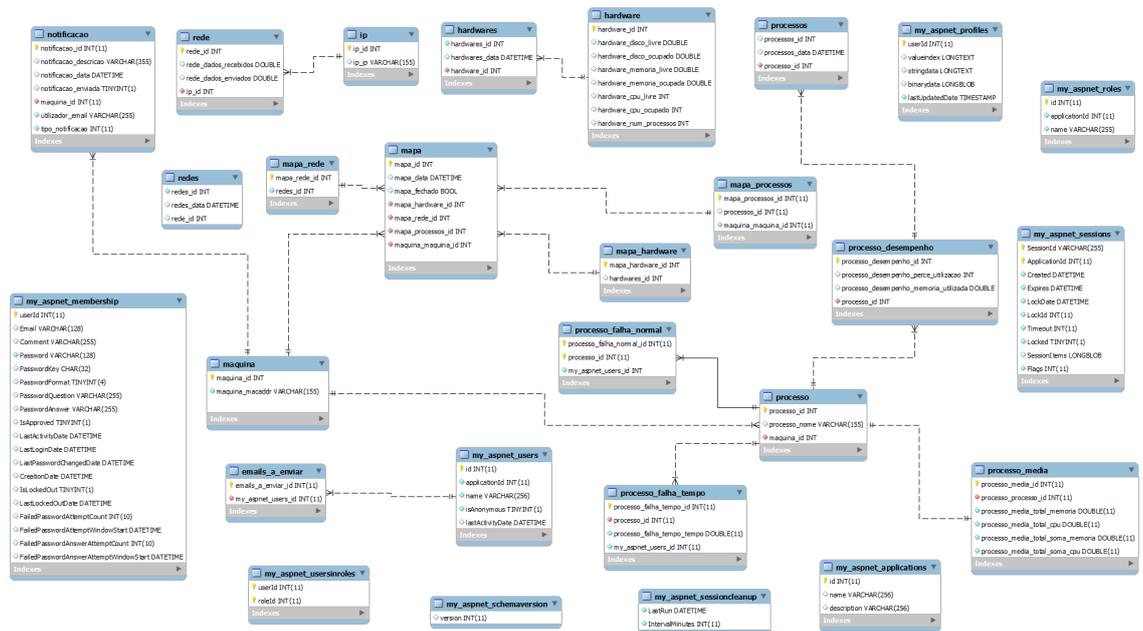


Figura 5.14: Esquema relacional da base de dados desenvolvida para suporte a plataforma



## Capítulo 6

# Análise do Produto - Simulação e Caso de Teste

### 6.1 Introdução

Para estudo e teste da aplicação desenvolvida procedeu-se à construção de um ambiente de simulação onde era pretendido pôr à prova a plataforma aquando de uma situação real de monitorização de máquinas. Durante a construção da aplicação, procedeu-se a vários testes, os quais eram feitos fazendo uso da máquina local, através do uso de um servidor *localhost*. De fato, esta situação não foi esclarecedora quanto à eficiência da total funcionalidade da plataforma na monitorização de outras máquinas, não *localhost*. Decidiu-se, portanto, aplicar alguma carga aos componentes de várias máquinas a fim de se avaliar a capacidade de resposta dada pela plataforma de monitorização quando sujeita a condições mais próximas da realidade. Para tal, reuniu-se o maior número de equipamentos (computadores) possível ligando os mesmos à rede local e por conseqüente à Internet, de forma a simular uma determinada rede de máquinas de uma instituição. Neste caso em concreto, o pretendido seria num ambiente hospitalar.

## 6.2 Cenário

Posto isto, as condições, os requisitos e as razões para as quais se efetuou esta simulação foram:

### **Nível de Instrumentação**

- Gestão de Informação:
  - Capacidade de monitorização de todos os dispositivos, respetivo desempenho e processos envolvidos.
- Protocolos de Comunicação:
  - Recolha de informação que deverá ser orientada à monitorização de todos os dispositivos, de modo a perceber qual o equipamento ativo para a recolha de dados referentes aos mesmos;
  - Flexibilidade de acesso, de forma a permitir suporte para gestão remota de máquinas por parte do utilizador.

### **Nível Aplicacional**

- Sistema Operativo base:
  - Microsoft Windows.
- Funcionalidades da aplicação:
  - Verificar se tem capacidade de monitorizar vários sistemas em tempo real.
  - Funcionalidades de segurança – login apenas permitido a utilizadores registados sem que haja fuga de informação.
  - Funcionalidades para gestão de notificações, como agendar e registar.
  - Verificar o envio de vários alertas para os utilizadores, aquando de uma determinada falha verificada.
- Desempenho:

- Tempo de resposta do sistema, ou seja verificar atraso verificado entre o momento de uma alteração e a chegada da mesma informação ao plataforma de monitorização;
- Requisitos de memória;
- Eficiência;

#### Nível do Utilizador

- Usabilidade:
  - Facilidade de utilização e flexibilidade ao nível da das diferentes unidades que constituem o sistema;
- Visualização:
  - Gestão através da interface *Web*.

### 6.3 Procedimento Efetuado

Antes de se passar à fase de execução das unidades pertencentes à plataforma de monitorização, foi necessário, numa primeira fase, a instalação de componentes essenciais para o funcionamento da plataforma. Instalou-se, então, a versão mais atual do *MySql Connector* e, em alguns dispositivos, foi necessário a atualização *.Net Framework* para uma versão mais recente.

Depois de cumpridos e verificados os passos anteriores em todas as máquinas iniciou-se o procedimento seguinte, que englobou a execução da unidade de análise em cada uma das máquinas, recolhendo os dados das mesmas. Continuamente, passou-se à execução da unidade de processamento, para processar e controlar os dados armazenados. Por fim procedeu-se à utilização e interação com a aplicação *web* para gestão, análise e interação com a informação recolhida das máquinas.

Comprovou-se ser possível visualizar a lista de todas as máquinas que estavam a ser monitorizadas, diferenciando-se as que estavam ligadas das desligadas, com a respetiva identificação.

Como seria de esperar, também se conseguiu visualizar, em tempo real, o desempenho dos componentes de cada um dos computadores ligados (processador, disco e memória RAM), assim como o histórico relativo a cada um deles. A opção de visualização dos processos em execução também foi conseguida com sucesso, efetivando-se a visualização na aplicação *web* de cada um dos processos a correr nas máquinas monitorizadas. Quanto aos gráficos relativos aos dados de Internet, também foi possível perceber o seu funcionamento, informando de forma correta o tráfego de dados recebidos e enviados por cada máquina.

Procedeu-se de seguida ao teste da central de notificações. Para tal, fez-se o agendamento de um alerta para uma das máquinas de modo a testar se iria ser recebido um alerta quando um determinado processo falhasse. O teste foi cumprido, usando o processo "notepad.exe" e o resultado foi positivo. A plataforma enviou um alerta sob a forma de e-mail para o utilizador aquando da falha propositada, fazendo ao mesmo tempo um registo na Lista de Notificações.

Também se testou o caso de aviso ao administrador aquando do restabelecimento de uma situação anómala. Depois de o utilizador ter recebido um email relativo à falha de um processo, voltou a executar-se esse mesmo processo. Nesta situação, a plataforma percebeu o restabelecimento automático da situação e informou o utilizador que o processo que tinha falhado, entretanto, voltou a funcionar.

De notar que as máquinas usadas na simulação de utilização foram: quatro máquinas virtuais, duas delas a correrem no mesmo computador, dois *desktop* e ainda três computadores portáteis individuais, perfazendo um total de nove máquinas a serem monitorizadas instantaneamente pela plataforma de monitorização.

## 6.4 Conclusão da Simulação

Toda a implementação e interação da plataforma de monitorização com as máquinas monitorizadas teve resultados de sucesso, não se verificando erros ou falhas na aplicação desenvolvida durante as operações efetuadas. A

unidade de análise mostrou ser um executável sem carga significativa para o sistema onde está a correr, tal como era esperado e pretendido.

De um modo geral, a plataforma apresentou um bom desempenho, o que significa que terá um comportamento semelhante quando estiver em funcionamento com um maior número de máquinas, num possível cenário futuro. Tanto a unidade de processamento como a interface *web* deram uma resposta ao nível do esperado, cumprindo as suas funções.

Um aspeto verificado e que requeria alguma melhoria seria na eficiência da unidade de análise, que apresenta alguma lentidão na altura de escrever em na base de dados os processos em execução, o que atrasa significativamente os tempos de produção de alertas.

Em relação a facilidade de uso e flexibilidade do sistema os resultados não poderiam ser mais animadores, pois o *website* apresenta um aspeto simples e a navegação é intuitiva, sem algum tipo de complexidade associado. As janelas têm um aspeto limpo e simples, e apresentam descrições bem visíveis para auxílio na gestão das atividades. Revelou-se, assim, um sistema *user friendly*.



# Capítulo 7

## Conclusões e Trabalho Futuro

### 7.1 Conclusões

Como foi perceptível ao longo deste documento, este projeto marcou o início da construção de um sistema que se acredita ser inovador e com grande utilidade na sua área de aplicação. O principal objetivo foi introduzir uma acentuada melhoria no que às tecnologias e sistemas de informação diz respeito, e, acima de tudo, no que se relaciona com os sistemas de monitorização.

É fundamental perceber que, com este trabalho, não se pretendeu apresentar uma solução de monitorização final, mas sim apresentar um protótipo de funcionamento para estudo e evolução futura, com o intuito de abrir novos caminhos para uma nova era de interação com os sistemas de informação e, continuamente, conduzir a um conhecimento superior, com uma grande margem de progressão e oportunidades de trabalho associadas.

Uma das principais metas para a elaboração deste trabalho foi claramente atingida, relacionando-se com a estruturação e elaboração de toda a aplicação para o funcionamento básico que era desejado: a monitorização de máquinas e a produção de alertas em caso de falha. As diferentes subunidades criadas, tanto a unidade de análise como a de processamento, estão de acordo com os projetos criados para as mesmas, cumprindo ambas a sua funcionalidade.

São aplicações inteiramente compatíveis com os sistemas para as quais foram projetadas, contribuindo, assim, para uma interoperabilidade de sucesso. Do ponto de vista computacional, são de execução leve e correm em ambiente *background*, ou seja não têm influência nem carga considerável no desempenho das máquinas que lhes oferecem suporte.

Em relação à ferramenta de interface gráfica, baseada na web para interação com o utilizador, os objetivos também foram atingidos. Trata-se de uma aplicação intuitiva e equipada com páginas visualmente e funcionalmente atrativas, o que facilita na utilização e manipulação, quer para visualização, quer para interação com a informação. A boa e eficiente comunicação entre as várias subunidades desenvolvidas e a reduzida necessidade de introdução de dados, facilita e permite a esta plataforma ser praticamente autónoma no seu funcionamento, podendo assim ser utilizada por indivíduos não especialistas em informática. Denote-se que era necessário cumprir com ambos os objetivos.

Num trabalho deste género, além de uma componente tecnológica, que passa pelo desenvolvimento de uma aplicação que visa a evolução e a diminuição das necessidades dos seus utilizadores, é exigido também um trabalho *background* de investigação. Este necessita de ter uma grande vertente teórica, de modo a dar suporte e, de certo modo, transmitir ao programador perceber quais os campos onde existe mais necessidade de expandir a tecnologia que se pretende desenvolver. Resumindo, o conhecimento retirado do trabalho de investigação é aplicado na tecnologia criada contribuindo para a resolução de problemas reais, os quais são encontrados todos os dias no quotidiano dos profissionais.

No que respeita à componente de investigação, as principais conclusões relacionam-se com o emergente crescimento das TI – Tecnologias de Informação, que surgem em consequência do crescimento da quantidade de dados armazenados e dos computadores de suporte. Por conseguinte, há necessidade de encontrar soluções para a gestão e monitorização das complexas infraestruturas de sistemas atualmente existentes. Nas organizações modernas e, num caso com especial atenção, como os ambientes hospitalares, dependem das tecnologias e mecanismos de gestão e monitorização para assegurarem o bom

desempenho e atividade contínua dos serviços que disponibilizam. Foi também elaborado um estudo de vários conceitos, como Sistemas de Informação, Interoperabilidade e Descoberta de Conhecimento em Bases de Dados, tendo sido dada particular ênfase ao entendimento teórico, para a sua aplicação na resolução de problemas associados ao contexto da plataforma.

Em relação ao estado de desenvolvimento e caracterização da plataforma de monitorização, conclui-se que a sua base é claramente as tecnologias associadas à Internet, aliando as potencialidades dos Sistemas de Gestão de Bases de Dados às facilidades de comunicação e interação através da Internet.

Como já referido, este sistema apresenta-se como um protótipo que cumpre os requisitos inicialmente especificados e planeados, o que demonstra cumprimento dos objetivos previamente definidos. Destacam-se

- Ser composto por subsistemas que em coordenação permitem gerir e lidar de forma eficiente com a informação de máquinas, assim como, fazer monitorização das mesmas;
- Trata-se de um sistema simples e funcional preparado para ser aplicado em sistemas operativos acessíveis (Windows);
- Cada uma das subunidades que constituem este sistema são completamente acessíveis, tanto ao utilizador como para os sistemas, no que diz respeito à carga exigida;
- Possui uma interface web *user friendly* e acessível ao utilizador em qualquer parte;
- Sendo um protótipo, oferece a possibilidade de se operar ao nível do código para uma futura modificação, caso se pretenda uma nova interação com outros subsistemas.

## 7.2 Trabalho Futuro

Na medida em que este projeto se trata de um protótipo, e é o início do desenvolvimento de um projeto que se acredita com capacidade evolutiva, irá

requerer algum trabalho futuro. Neste subcapítulo apresentam-se soluções, atualmente já pensadas, que seriam de enorme valorização.

Assim, sabendo as atuais funcionalidades e características da aplicação, a grande evolução passaria por aplicar um conceito de *Mining*, tornando a aplicação num sistema com Inteligência Artificial com capacidade de previsão e assim antecipar de forma eficiente uma falha num sistema. O *Mining* seria feito aos dados recolhidos e armazenados, *data mining*, conseguindo-se assim perceber padrões por parte de dados passados e retirar conclusões antecipadas para uma situação futura que se viesse a verificar.

Apesar de a unidade de análise só estar preparada para aplicação em Windows, uma outra alteração com grande importância estaria relacionada com a sua compatibilidade com outros sistemas operativos, como MacOS ou Linux, de forma a aumentar as várias frentes de aplicação desta plataforma.

De acordo com o tema proposto, a finalidade da plataforma seria para monitorização das máquinas de um hospital no Norte do País. Para tal, terá ainda de se perceber e estudar as condições existentes nos sistemas desse hospital para fazer as alterações necessárias na estrutura desenvolvida, de forma a se estabelecer uma ligação interoperável entre os vários componentes.

Por fim, seria importante aumentar a eficiência da unidade de análise aquando da recolha e escrita dos dados na base de dados. Na escrita dos processos que estão em execução numa determinada máquina, a aplicação mostra alguma lentidão. A solução passaria pela revisão e optimização do código escrito.

Se efetivamente, num futuro, as etapas acima descritas forem elaboradas, a plataforma apresentará todas as características e funcionalidades que no início se imaginou concretizar, estando assim apta a resolver os problemas quotidianos com que os utilizadores atualmente mais se deparam.

*Gestão + Monitorização = Prevenção e resolução de erros e  
falhas*

# Bibliografia

- [1] Goretí Marreiros, Ricardo Santos, Carlos Ramos, José Neves, Paulo Novais, José Machado, and José Bulas-Cruz. Ambient intelligence in emotion based ubiquitous decision making.
- [2] António Abelha, Cesar Analide, José Machado, José Neves, Manuel Santos, and Paulo Novais. Ambient intelligence and simulation in health care virtual scenarios. *Departamento de Informática, Universidade do Minho, Braga, Portugal; Departamento de Sistemas de Informação, Universidade do Minho, Guimarães, Portugal.*
- [3] Microsoft.net Development Series. Visual studio development environment model, January 2008. URL [http://msdn.microsoft.com/en-us/library/bb165114\(v=vs.80\).aspx](http://msdn.microsoft.com/en-us/library/bb165114(v=vs.80).aspx).
- [4] *Windows Management Instrumentation*. Microsoft Corporation, July 2012. URL [http://msdn.microsoft.com/en-us/library/windows/desktop/aa394582\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa394582(v=vs.85).aspx).
- [5] Microsoft Corporation. Overview of ado.net - .net framework 2.0.
- [6] Microsoft Corporation. Visual web developer express - guided tour, 2005. URL <http://www.asp.net/Guided-Tour/s15.aspx>.
- [7] Elizabeth Hardcastle. *Business Information Systems*. Elizabeth Hardcastle & Ventus Publishing APS, 2008.
- [8] Miguel Miranda, Júlio Duarte, António Abelha, José Machado, José Neves, and João Neves. Interoperability in healthcare. *Universidade do*

*Minho, CCTC, Departamento de Informática, Braga, Portugal; Centro Hospitalar de Vila Nova de Gaia e Espinho, Portugal.*

- [9] José Machado, Victor Alves, António Abelha, and José Neves. Ambient intelligence via multiagent systems in the medical arena. .
- [10] António Carlos da Silva Abelha. *Sistemas Multiagente como Suporte a Trabalho Cooperativo em Unidades Hospitalares*. PhD thesis, Universidade do Minho, Departamento de Informática, Escola de Engenharia, Universidade do Minho, Campus de Gualtar, 4710, Braga Codex, Portugal, 2004.
- [11] Hermano Filipe Domingues Pereira. *Avaliação de ferramentas de monitorização e gestão de redes*. PhD thesis, Departamento de Electrónica, Telecomunicações e Informática - Universidade de Aveiro, Portugal Informática, 2009.
- [12] Chih-Lin Chia, W. Nick Streetb, and Marcia M. Wardc. Building a hospital referral expert system with a prediction and optimization-based decision support system algorithm. *Journal of Biomedical Informatics*, April 2008.
- [13] T G Thompson and D J Brailer. The decade of health information technology: delivering consumer-centric and information-rich health care: Framework for strategic action. *Washington, DC: Department of Health and Human Services, National Coordinator for Health Information Technology*, March 2007.
- [14] José Neves, Manuel Santos, José Machado, António Abelha, Sollari Allegro, and Maria Salazar. Electronic health records - organizational, regional, national, or worldwide. *Departamento de Informática, Departamento de Sistemas de Informação, Universidade do Minho, Portugal and Centro Hospitalar do Porto, EPE Porto, Portugal.*
- [15] José Machado, Miguel Miranda, Pedro Gonçalves, António Abelha, José Neves, and Alberto Marques. Aidatrace - interoperation platform for active monitoring in healthcare environments. .

- [16] Eleni Mangina. Intelligent agent-based monitoring platform for applications in engineering. *University College Dublin Department of Computer Science Belfield, Dublin, Ireland*, 2005.
- [17] V. Bicer, O. Kilic, A. Dogac, and G. Laleci. Archetype-based semantic interoperability of web service messages in the healthcare domain. *International Journal on Semantic Web & Information Systems*, 2005.
- [18] Hugo Peixoto, José Machado, and António Abelha. Interoperabilidade e o processo clínico semântico.
- [19] André Elias de Onofre Ferreira Lima. *Sistema de monitorização de aplicações e equipamentos de portagem*. PhD thesis, INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA, 2008.
- [20] CETIS. Cetus: Interoperability standards and assessment.
- [21] Fernando Emmanuel da Costa Pereira Nunes. *Infra-estrutura de Monitorização Activa de Sistemas Informáticos*. PhD thesis, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, Setembro 2009.
- [22] J. Cucurull, R. Martí, G. Navarro-Arribas, S. Robles, and J. Borrell. Full mobile agent interoperability in an ieee-fipa context. *The Journal of Systems and Software*, Junho 2009.
- [23] Patricia Gibbons, Noam Arzt, Susie Burke-Beebe, Chris Chute, Gary Dickinson, Tim Flewelling, Thomas Jepsen, Don Kamens, Joanne Larson, John Ritter, Michael Rozen, Sherry Selover, and Jean Stanford. Scoping interoperability for health care. Health Level Seven - EHR Interoperability Work Group, February 2007.
- [24] José Machado, Miguel Miranda, António Abelha, José Neves, and João Neves. Modeling medical ethics through intelligent agents. .
- [25] Angelo Rossi-Mori. Computer-based handling of clinical information.

- [26] H. Rigor, J. Machado, A. Abelha, J. Neves, and C. Alberto. A web-based system to reduce the nosocomial infection impact in healthcare units. *Webist. Madeira, Portugal*, 2008.
- [27] José Machado, António Abelha, Paulo Novais, João Neves, and José Neves. Quality of service in healthcare units. 2008.
- [28] Natália Botica, Maribel Yasmina Santos, and Francisco Sande Lemos. Data mining e património arqueológico. *Conferência IADIS Ibero-Americana WWW/Internet 2003*, 2003.
- [29] U.M. Fayyad, G. Piatetsky-Shapiro, S. Smyth, and R. Uthurusamy. Advances in knowledge discovery and data mining. *MIT Press*, 1996.
- [30] Armando Cruz and Paulo Cortez. Data mining via redes neuronais artificiais e máquinas de vectores de suporte. *Revista de Estudos Politécnicos Polytechnical Studies Review 2009, Vol VII, no 12, 099-118*, 2009.
- [31] Armando Jorge Ribeiro da Cruz. *Data Mining via Redes Neuronais Artificiais e Máquinas de Vectores de Suporte*. PhD thesis, Escola de Engenharia - Universidade do Minho, 2007.
- [32] Ian H. Witten and Eibe Frank. *Data Mining - Practical Machine Learning Tools and Techniques*. Diane Cerra, second edition, 2005.
- [33] A. M. Cisterl and N. F. F. Ebecken. *CRM through DM: a case study*.
- [34] M. F. Santos and C. Azevedo. Data mining, descoberta de conhecimento em bases de dados. FCA – Editora de Informática, 2005.
- [35] S. Nicholson. The bibliomining process: Data warehousing and data mining for library decision-making. *Information Technology and Libraries* 22, 2003.
- [36] IEEE Standards Association. *Standard Group MAC Addresses: A Tutorial Guide*. IEEE, September 2011.

- [37] Andrea De Lucia, Filomena Ferrucci, Genny Tortora, and Maurizio Tucci. *Emerging Methods, Technologies and Process Management in Software Engineering*. February 2008.
- [38] Craig Larman and Victor R. Basili. Iterative and incremental development: A brief history. *IEEE Computer Society*, 2003.
- [39] Lawrence Bernstein and C. M. Yuhás. *Trustworthy Systems Through Quantitative Software Engineering*. John Wiley & Sons, Inc., Hoboken, New Jersey, October 2005.
- [40] Tore Dyba Åä and Torgeir Dingsøy. Empirical studies of agile software development: A systematic review. January 2008.
- [41] Anders Hejlsberg, Scott Wiltamuth, and Peter Gold. *The C# Programming Language*. Second edition.
- [42] Richard Blum. *C# Network Programming*. SYBEX, 2003.
- [43] Don Gosselin. *ASP.Net Programming with C# & SQL Server*. 2009.
- [44] Jesse James Garrett. Ajax: A new approach to web applications. February 2005.
- [45] Philippe Le Hégarét. Document object model (dom), January 2005. URL <http://www.w3.org/DOM/#what>.
- [46] Baron Schwartz, Peter Zaitsev, and Vadim Tkachenko. *High Performance MySQL: Optimization, Backups, and Replication*. O'Reilly Media, Inc., third edition, March 2012.
- [47] José Manuel Ferreira Machado. *Bases de Dados*. Universidade do Minho, 2009.
- [48] Google Developers. Introduction to using chart tools.
- [49] Jesse Liberty, Dan Maharry, and Dan Hurwitz. *Programming ASP.NET 3.5*. O'Reilly Media, Inc., October 2008.

- [50] G. W. Hansen and J. V. Hansen. *Database management and design*. Upper Saddle River, NJ, USA, second edition, 1996.

# Apêndice A

## Manual de Utilizador

MANUAL DE UTILIZAÇÃO DA  
PLATAFORMA DE MONITORIZAÇÃO  
DE SISTEMAS INFORMÁTICOS



# Conteúdo

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Apresentação</b>                         | <b>7</b>  |
| 1.1      | Registo . . . . .                           | 7         |
| 1.2      | Autenticação . . . . .                      | 8         |
| 1.3      | Home page . . . . .                         | 8         |
| 1.3.1    | Barra de Menu . . . . .                     | 9         |
| <b>2</b> | <b>Monitorização</b>                        | <b>11</b> |
| 2.0.2    | Monitorização atual . . . . .               | 11        |
| 2.0.3    | Histórico dos dados Monitorizados . . . . . | 16        |
| <b>3</b> | <b>Notificações</b>                         | <b>19</b> |
| 3.0.4    | Agendar notificação . . . . .               | 19        |
| 3.0.5    | Lista de Notificações . . . . .             | 20        |



# Lista de Figuras

|     |   |    |
|-----|---|----|
| 1.1 | Registo de um utilizador . . . . .                                | 7  |
| 1.2 | Autenticação do Utilizador . . . . .                              | 8  |
| 1.3 | Página principal depois de um utilizador se autenticar . . . . .  | 9  |
| 1.4 | Barra de menu para acesso aos diferentes conteúdos . . . . .      | 9  |
| 2.1 | Lista de máquinas e suas informações . . . . .                    | 12 |
| 2.2 | Página de seleção das opções de monitorização . . . . .           | 13 |
| 2.3 | Processos que estão em execução na máquina . . . . .              | 14 |
| 2.4 | Gráficos dinâmicos para processador e RAM . . . . .               | 14 |
| 2.5 | Gráficos estáticos para Processador e RAM e Disco . . . . .       | 15 |
| 2.6 | Gráfico monitorização atual para os Dados de Rede . . . . .       | 16 |
| 2.7 | Gráficos para o dados históricos . . . . .                        | 17 |
| 3.1 | Página de agendamento de notificações . . . . .                   | 19 |
| 3.2 | Página da lista de notificações enviadas pela plataforma. . . . . | 20 |

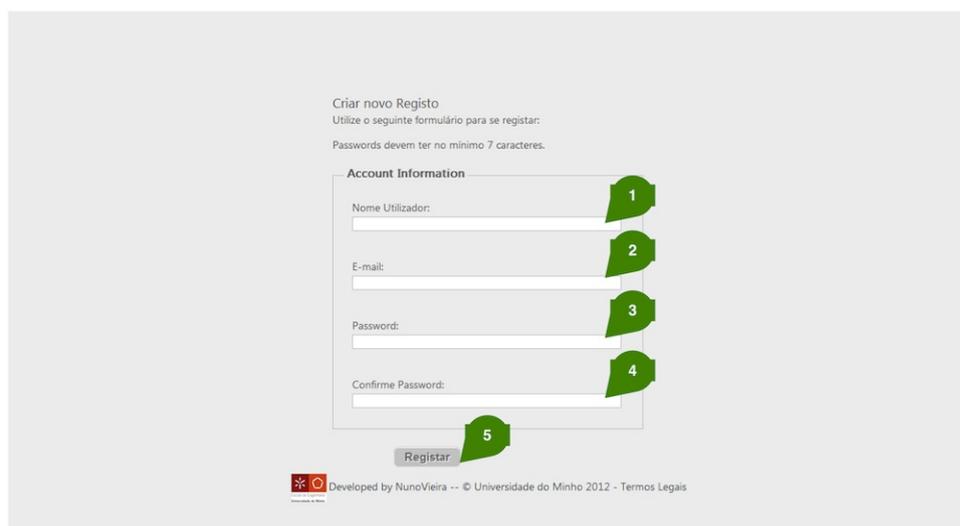


# Capítulo 1

## Apresentação

Este manual servirá como um guia que facilite a utilização e interação dos utilizadores com a plataforma. Serão apresentados todas os casos possíveis de uso que um futuro utilizador poderá encontrar.

### 1.1 Registo



Criar novo Registo  
Utilize o seguinte formulário para se registar:  
Passwords devem ter no mínimo 7 caracteres.

**Account Information**

Nome Utilizador: 1

E-mail: 2

Password: 3

Confirme Password: 4

Registar 5

Developed by NunoVieira -- © Universidade do Minho 2012 - Termos Legais

Figura 1.1: Registo de um utilizador

1. Introduzir um nome de utilizador
2. Introduzir um email válido

3. Introduzir uma password
4. Confirmar password
5. Validar Registo

## 1.2 Autenticação

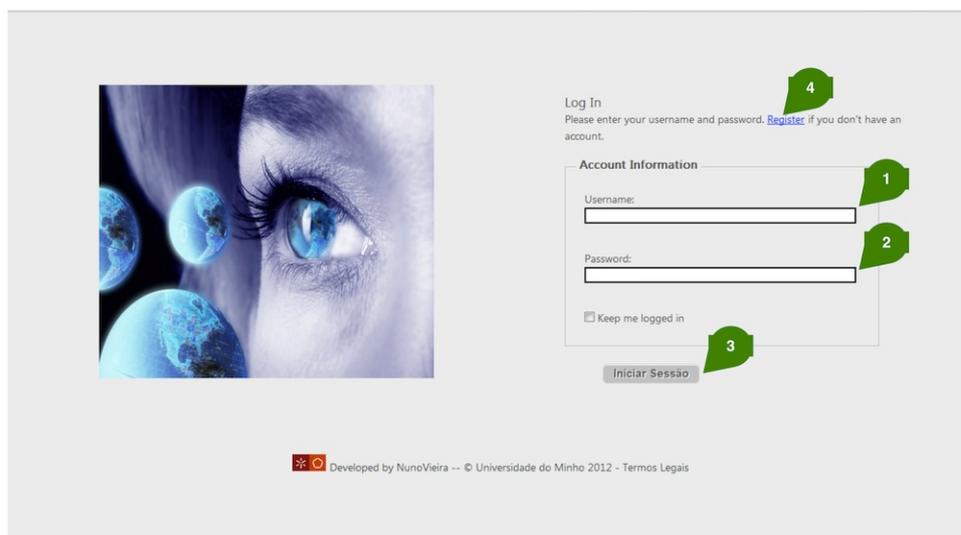


Figura 1.2: Autenticação do Utilizador

Depois de ter efetuado registo utilizador terá acesso ao menu de autenticação.

1. Introduzir nome de utilizador
2. Introduzir palavra-passe
3. Validar autenticação
4. Entrar no menú de registo de novo utilizador

## 1.3 Home page

Depois de se autenticar utilizador encontrará na Home Page:

1. Barra de Menu para acesso às funcionalidades da plataforma



Figura 1.3: Página principal depois de um utilizador se autenticar

2. Botão Logout para terminar sessão
3. Zona de *banners* para introdução de novidades e notícias
4. Nome e id do utilizador autenticado

### 1.3.1 Barra de Menu

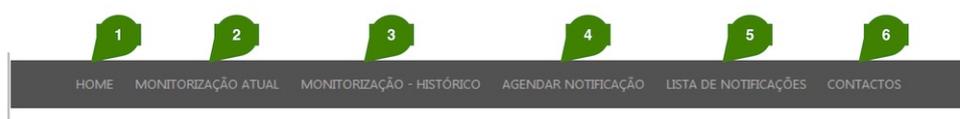


Figura 1.4: Barra de menu para acesso aos diferentes conteúdos

1. Ver Home Page
2. Ver Monitorização atual

3. Ver histórico da monitorização
4. Acesso à pagina para agendar notificações
5. Acesso à Lista de Notificações
6. Ver a página de contactos

# Capítulo 2

## Monitorização

Dentro do capítulo de monitorização, como já visto, existe possibilidade de acesso a:

### 2.0.2 Monitorização atual

Para visualizar o desempenho atual primeiro terá que escolher a máquina que pretende ver da lista de máquinas.

The image shows a monitoring interface. At the top, a detailed view of a machine's status is shown, with a green callout '2' pointing to it. Below this is a section titled 'LISTA DE MÁQUINAS MONITORIZADAS' with a green callout '1' pointing to the list. The list contains four entries, each with a monitor icon and a text box containing machine details. A large green arrow points from the first entry in the list to the detailed view above.

**Estado da Máquina:**  
Máquina Desligada  
MACADDRESS: 00:0C:29:3D:A7:C6  
Numero de Processos em Curso:  
Memória RAM usada:1525 Mb  
Disco Usado:33 Gb  
Processador em uso:0 %

**LISTA DE MÁQUINAS MONITORIZADAS**  
Selecione a máquina para a qual deseja visualizar o desempenho atual:

| Estado da Máquina: | MACADDRESS:       | Numero de Processos em Curso: | Memória RAM usada: | Disco Usado: | Processador em uso: |
|--------------------|-------------------|-------------------------------|--------------------|--------------|---------------------|
| Máquina Desligada  | 00:0C:29:3D:A7:C6 |                               | 1525 Mb            | 33 Gb        | 0 %                 |
| Máquina Desligada  | 00:0C:29:00:F8:BE |                               | 438 Mb             | 13 Gb        | 0 %                 |
| Máquina Desligada  | 00:23:6C:A9:71:E2 |                               | 963 Mb             | 229 Gb       | 4 %                 |
| Máquina Desligada  | 00:50:56:E6:18:E3 |                               | 1700 Mb            | 33 Gb        | 15 %                |

Figura 2.1: Lista de máquinas e suas informações

1. Lista das máquinas monitorizadas

2. Resumo de informação relativo a cada uma delas.

Depois de selecionar uma máquina utilizador deverá escolher o que pretende monitorizar.



Figura 2.2: Página de seleção das opções de monitorização

1. Ver Processos
2. Ver Hardware
3. Ver dados de Rede

Em 4 o utilizador tem informação da máquina que selecionou.

### Processos

1. Nome do Processo
2. Processador utilizado por cada processo
3. Memória utilizada por cada processo
4. Campo de pesquisa de processos

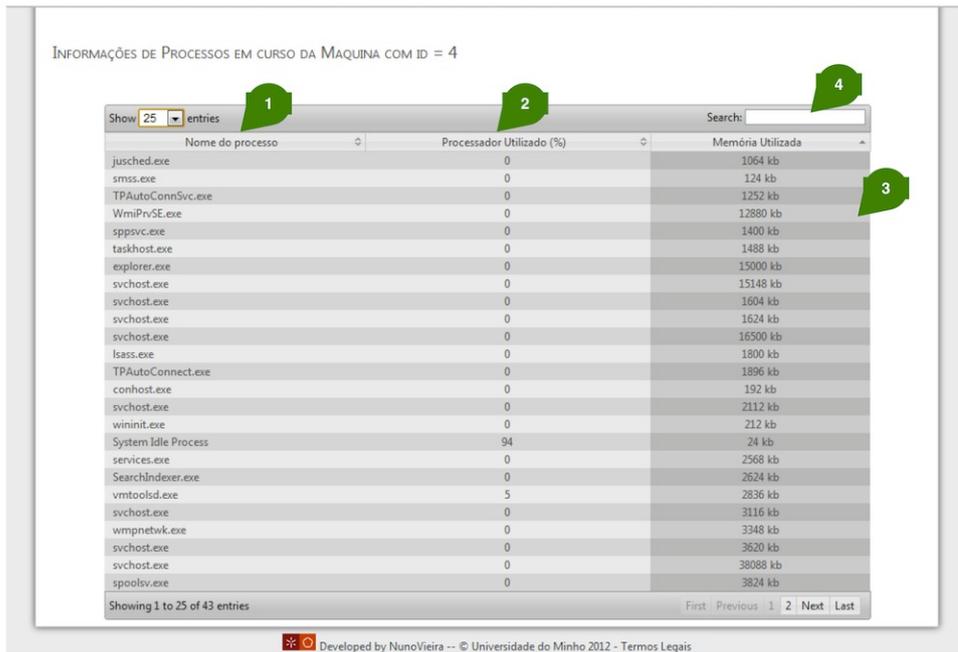


Figura 2.3: Processos que estão em execução na máquina

## Hardware

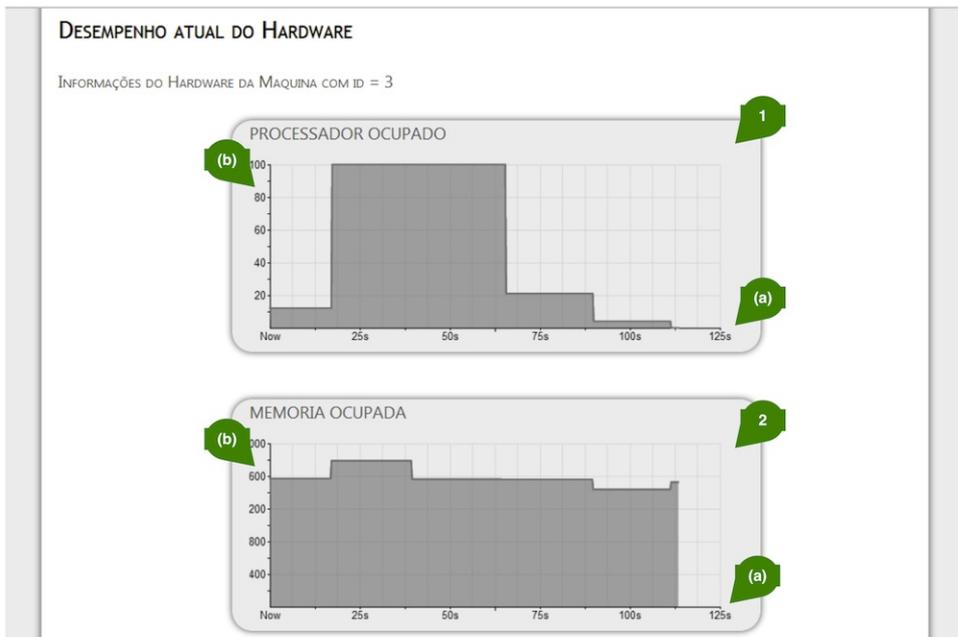


Figura 2.4: Gráficos dinâmicos para processador e RAM

1. Gráfico dinâmico para o desempenho do processador
  - (a) Tempo
  - (b) Percentagem de uso
2. Gráfico dinâmico para o desempenho da RAM
  - (a) Tempo
  - (b) Espaço ocupado (Mb)

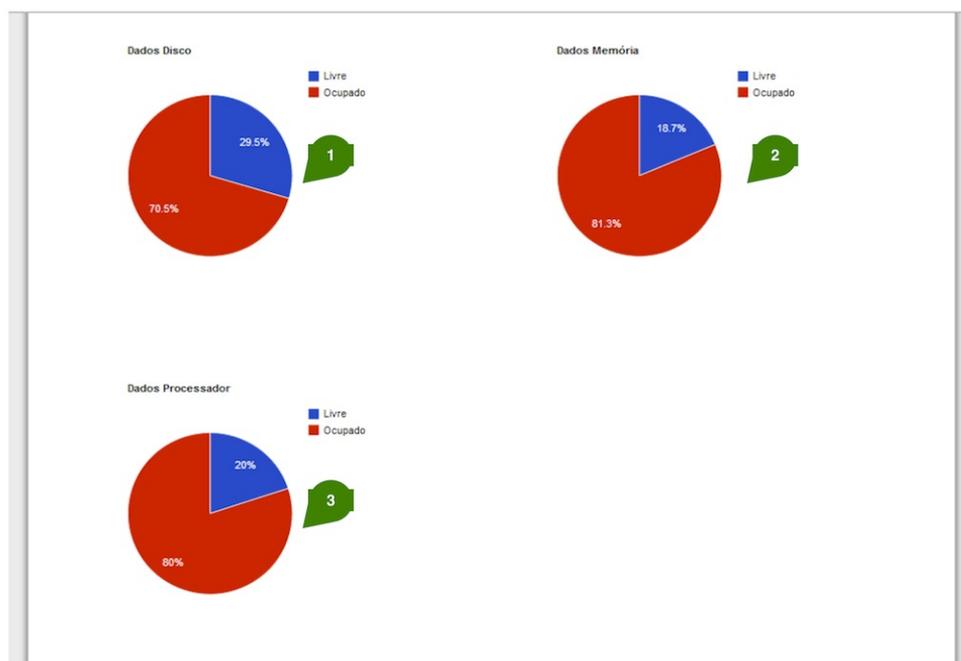


Figura 2.5: Gráficos estáticos para Processador e RAM e Disco

1. Percentagem livre e ocupado para Disco
2. Percentagem livre e ocupado para RAM
3. Percentagem livre e ocupado para Processador

## Rede

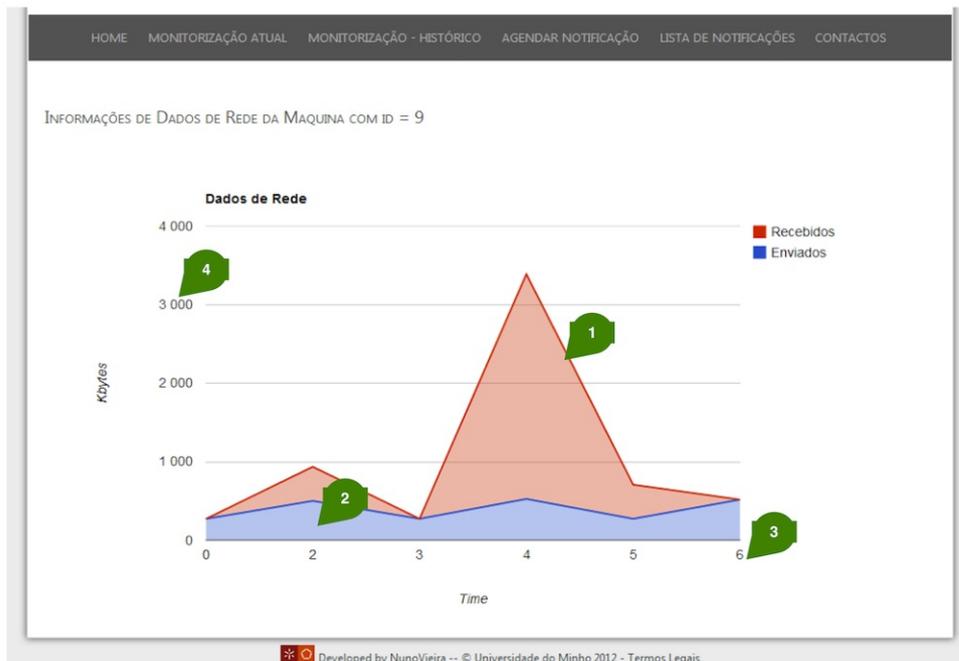


Figura 2.6: Gráfico monitorização atual para os Dados de Rede

1. Dados Recebidos
2. Dados Enviados
3. Tempo
4. Tráfego (Kb)

### 2.0.3 Histórico dos dados Monitorizados

Se o utilizador optar por ver os dados monitorizados, terá que seleccionar na barra de Menu, **Monitorização-Histórico**.



Figura 2.7: Gráficos para o dados históricos

1. Gráfico dos dados históricos para o processador
2. Gráfico dos dados históricos para a memória RAM
3. Para o gráfico do processador detalhes do comportamento para as datas escolhidas
4. Comportamento geral do processador desde o início da sua monitorização, com as datas para seleção temporal



# Capítulo 3

## Notificações

Para gestão das notificações utilizador poderá optar por agendar notificações /alertas para ser notificado ou escolher ver a lista de notificações já enviadas.

### 3.0.4 Agendar notificação

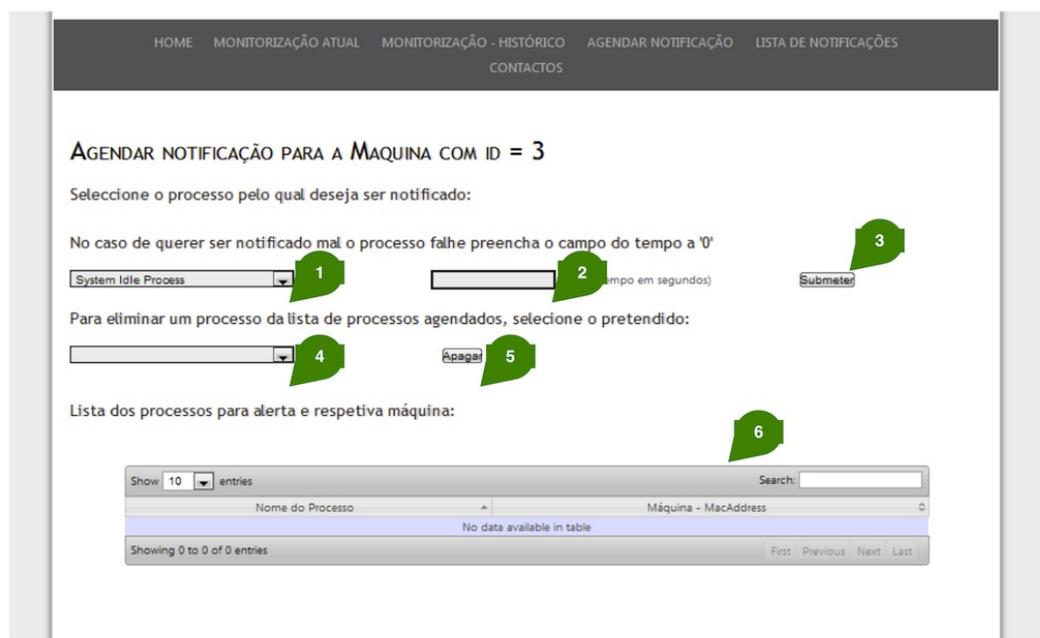


Figura 3.1: Página de agendamento de notificações

Para agendar uma notificação, utilizador terá que:

1. Selecionar o processo para o qual deseja ser notificado
2. Se for um processo temporário, preencher com o tempo de execução, caso contrário terá que preencher o campo a '0'
3. Submeter o agendamento
6. Tabela com os registos das notificações agendadas

Caso o utilizador pretenda remover um alerta, para não ser mais notificado terá que:

4. Selecionar o processo que se encontra agendado
5. Submeter a ação

### 3.0.5 Lista de Notificações

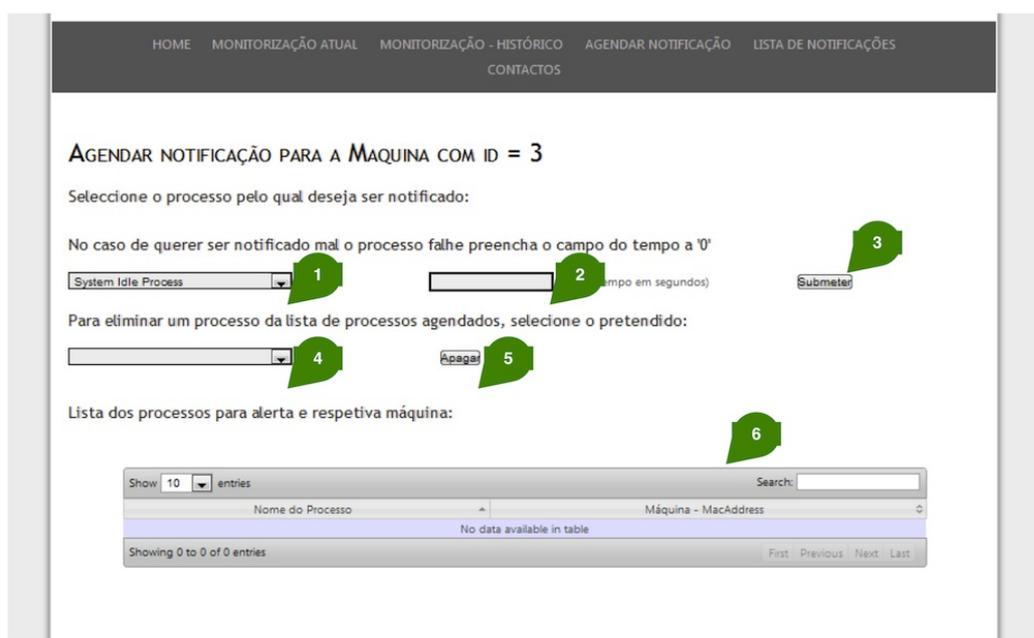


Figura 3.2: Página da lista de notificações enviadas pela plataforma.

Utilizador pode consultar:

1. ID da notificação enviada
2. Descrição da notificação
3. Data e hora da falha verificada
4. Máquina onde aconteceu a falha
5. Email do utilizador que recebeu o alerta
6. Tipo de notificação que foi enviada