

329

Universidade Federal de Minas Gerais

Faculdade de Ciências

Departamento de Matemática e Informática

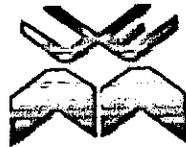
TRABALHO DE MONOGRAFIA

USO DAS TÉCNICAS DE SUPORTE DE DADOS PARA
MIGRAÇÃO DE SISTEMAS DE INFORMAÇÃO

ESTUDO DE CASO SISTEMA DE INFORMAÇÃO PARA GESTÃO DE
EQUIPAMENTOS MÉDICOS

IT-329

EURICO JOSÉ



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE MATEMÁTICA E INFORMÁTICA

TRABALHO DE LICENCIATURA

**USO DAS METODOLOGIAS ORIENTADAS A OBJECTOS PARA
MIGRAÇÃO DE SISTEMAS DE INFORMAÇÃO LEGADOS:**

**ESTUDO DE CASO SISTEMA DE INFORMAÇÃO PARA GESTÃO DE
EQUIPAMENTOS MÉDICOS**

EURICO JOSÉ



UNIVERSIDADE EDUARDO MONDLANE

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE MATEMÁTICA E INFORMÁTICA

TRABALHO DE LICENCIATURA

**USO DAS METODOLOGIAS ORIENTADAS A OBJECTOS PARA
MIGRAÇÃO DE SISTEMAS DE INFORMAÇÃO LEGADOS:
ESTUDO DE CASO SISTEMA DE INFORMAÇÃO PARA GESTÃO DE
EQUIPAMENTOS MÉDICOS**

ESTUDANTE: EURICO JOSÉ

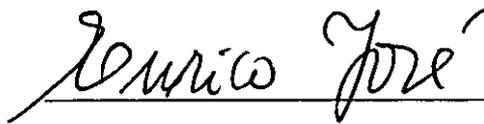
SUPERVISOR: PROF. DOUTOR JOSÉ LEOPOLDO NHAMPOSSA

MAPUTO, MAIO DE 2008

Declaração de Honra

Eu, Eurico José, declaro por minha honra, fidelidade e zelo que este trabalho é da minha autoria e é resultado das minhas próprias investigações e o mesmo foi realizado apenas para ser submetido como Trabalho de Licenciatura em Informática na Faculdade de Ciências da Universidade Eduardo Mondlane.

Maputo, 6 de Maio de 2008

A handwritten signature in black ink, reading "Eurico José", written over a horizontal line.

(Eurico José)

Dedicatória

A Deus.

Aos meus pais José Abibo e Aurissa Varieque.

À memória da minha tia Beatriz Varieque.

À memória dos meus irmãos Constantino José Abibo e Gildo José Abibo

Agradecimentos

Diversas individualidades singulares e institucionais contribuíram não só para realização deste trabalho como também me ajudaram durante a minha formação, a todos eles endereço os meus agradecimentos.

Aos meus pais que me deram a vida e me educaram para que eu fosse um homem digno a dimensão do que eles são, meus sinceros agradecimentos.

Sinceros agradecimentos ao meu supervisor Prof. Doutor José Leopoldo Nhampossa, pela orientação e acompanhamento durante a realização deste trabalho. Este agradecimento é extensivo ao Saugene, PhD pelas sugestões e seu acompanhamento não só como estudante mas também profissionalmente.

Muito obrigado a todo pessoal do Departamento de Manutenção do MISAU em especial ao Eng.º Nhambessa e Eng.º Abubacar Sumalgy pelo acompanhamento e seu apoio incondicional.

As minhas irmãs Irene e Fernanda, meus irmãos Jorge, Armando e Rodrigues que são fonte da minha inspiração, o meu muito obrigado.

A minha namorada Anatercia e ao meu filho Pacuneta, pela paciência, carinhos e amizade que têm me dado, o meu muito obrigado e que continuem assim.

Um agradecimento para toda minha família, em especial na pessoa de Tiorenço Nioveque pela paciência que tem tido em dar a sua ajuda moral, material em momentos precisos. Ao Florindo Paulo que dentro das suas possibilidades não teve mãos a medir em me ajudar.

Aos colegas da turma de Agosto 2002 do DMI, particularmente Valnir Fiel, Salvador Cardoso, Jorge Paulino, Márcia Juvane e Arlindo dos Reis (Docas) pelo apoio moral, material nas fases críticas e pelas suas sugestões e críticas desde a carteira até a fase de realização deste trabalho, o meu muito obrigado.

A todos meus amigos em especial Victorino, Henriques, Ergel e Guilhermino obrigado pela amizade.

Lista de Acrónimos

- B_Info – Boletins de Informação;
- BR_I – Boletim de Retro-Informação;
- CASE – Computer Aided Software Engineering;
- CLR – Common Language Runtime;
- COTEC – Comissão Técnica de Equipamentos e Consumíveis;
- DFD – Diagrama de Fluxo de Dados;
- DHIS – Distrital Helth Information System (Sistema de Informação para Saude Distrital);
- DM – Departamento de Manutenção;
- DMI – Departamento de Matemática e Informática;
- DNP – Direcção Nacional do Património;
- DPS – Direcção Provincial de Saúde;
- ECRI – Emergency Care Research Institute;
- FaPRE/OO – Família de Padrões de Reengenharia Orientada a Objectos;
- FO – Folha de Obra;
- GUI – Graphical User Interface (Interface de utilizador gráfica);
- HISP – Helth Information System Program (Programa de Sistema de Informação de Saúde) ;
- HP – Hewlett-Packrd;
- JDBC – Java Database Connectivity;
- MISAU – Ministério da Saúde;
- MySQL – My Structered Query Language;
- ODBC – Open Database Connectivity;
- OMT – Object Modeling Techique (Técnica de Modelação de Objectos);
- OO – Object Oriented (Orientado a Objectos) ;
- OOSE – Object Oriented Software Engineering (Engenharia de Software Orientado a Objectos);
- SDL – Specification Description Language (Linguagem de Especificação e Desenho) ;
- SGBD – Sistema de Gestão Base de Dados;
- SGEM – Sistema de Gestão de Equipamentos Médicos;
- SI – Sistemas de Informação;

- SIL – Sistemas de Informação Legados;
- SIS – Sistema de Informação para Saúde;
- SMH – Secção de Manutenção Hospitalar;
- SNS – Sistema Nacional de Saúde;
- SPM – Secção Provincial de Manutenção;
- SSADM – Structured Systems Analysis and Design Method;
- TI – Tecnologias de Informação;
- UEM – Universidade Eduardo Mondlane;
- UM – Unidade de Manutenção;
- UMDNS – Universal Medical Device Nomenclature System (Sistema Universal de Nomenclatura de Equipamento Médicos);
- UML – Unified Modeling Language (Linguagem de Modelação Unificada);
- US – Unidade Sanitária;
- VB.NET – Visual Basic .NET;
- VBA – Visual Basic for Applications;

Resumo

O presente trabalho apresenta uma análise informada teoricamente, empiricamente e prática sobre o uso de metodologias orientadas a objectos para a migração de sistemas de informação legados.

A base teórica é informada pelas metodologias orientadas a objectos, sistemas de informação legados e ferramentas de migração de sistemas de informação legados. O paradigma de orientação a objectos define princípios e conceitos que permitem o desenvolvimento de sistemas mais próximos da realidade visual. Neste trabalho, descreve-se as diferentes metodologias de análise e de desenho orientados a objectos com maior ênfase na metodologia *Object Oriented Software Engineering* (OOSE) e das ferramentas que auxiliam no processo de migração de sistemas legados.

A base empírica é o estudo de caso de sistema de informação para gestão de equipamentos médicos e o foco do trabalho é a migração do sistema legado conhecido como SIM-ORGANIZER. Trata-se de um estudo qualitativo, exploratório e descritivo, sem carácter generalizador com o foco no processo e não no produto. O método de pesquisa usado, foi o “estudo de caso”. Este método sugere a análise indutiva dos dados e considera o local natural como fonte directa de dados.

A observação directa e participativa que se caracteriza pelo envolvimento do pesquisador nas actividades da organização cujo estudo se envolve, foram as técnicas usadas para a colheita de dados de suporte do trabalho combinadas num processo incremental e iterativo com as entrevistas e revisão de literatura.

A base prática é a modelação do sistema de informação para gestão de equipamentos médicos, usando a metodologia OOSE, Padrões de Reengenharia Orientados a Objectos, *Visual Studio 2005* e *MySQL*.

Este trabalho é uma contribuição teórico-prática para reengenharia de software para uso/implementação em organizações.

ÍNDICE

Conteúdo	Página
Declaração de Honra.....	i
Dedicatória.....	ii
Agradecimentos	iii
Lista de Acrónimos.....	iv
Resumo	vi
Capítulo I: Introdução e Objectivos	1
1.1. Introdução	1
1.2. Motivação	3
1.3. Definição do Problema.	3
1.4. Objectivos	5
1.4.1. Objectivo Geral.....	5
1.4.2. Objectivos Específicos.....	5
Capítulo II: Revisão de Literatura.....	6
2.1. Metodologias Orientadas a Objectos	6
2.1.1. O Paradigma Orientação a Objectos.....	7
2.1.2. Object Oriented Software Engineering (OOSE).....	9
2.1.2.1. Aspectos Gerais de OOSE	9
2.1.3.2. Técnicas de Modelagem para a Descrição de Sistema em OOSE	9
2.1.2.3. Elementos Sintácticos das Técnicas de Modelagem em OOSE	10
2.1.3. Grady Booch	13
2.1.4. Object Modeling Techique – Técnica de Modelação de Objectos (OMT).....	15
2.1.4.1. Fase de Análise	15
2.1.4.2. Fase de Desenho	16
2.1.4.3. Fase de Construção	17
2.1.5. Fusion.....	17

2.1.6. Coad-Yourdon	19
2.1.7. UML.....	20
2.1.8. Análise comparativa das metodologias orientadas a objectos	21
2.1.10. Razões para Escolher a Metodologia OOSE	23
2.2: Sistemas de Informação Legados	24
2.2.1. Perspectiva de Sistemas de Informação Legado	24
2.2.2. Processo de Transição em SIL.....	25
2.2.2.1. Formação e treinamento da Equipa	26
2.2.2.2. Organização da Base de Informações	26
2.2.2.3. Mapear Elementos de Diagramas de Fluxos de Dados para Casos de Uso na Metodologia Escolhida	27
2.2.3. Ferramentas para Migração de Sistemas de Informação Legados.....	27
2.2.3.1. Ferramentas CASE	27
2.2.3.2. Família de Padrões de Reengenharia Orientada a Objectos – FaPRE/OO	28
2.2.3.3. Sistemas de Gestão de Base de Dados (SGBD).....	29
2.2.3.4. O MySQL	30
2.2.3.5. Linguagens de Implementação	31
2.2.3.6 Tecnologia Visual Studio .Net.....	32
Capitulo III. Metodologia de Pesquisa.....	33
3.1. Cenário e Estudo de Caso	33
3.1.1. SIL no SIS em Moçambique.....	33
3.1.2. Abordagens Anteriores aos SIL do MISAU	35
3.2. Método	36
Capitulo IV: Modelação do Sistema de Informação para Gestão de Equipamentos Médicos.....	40
4.1. Descrição do sistema.....	40
4.2. Modelo de Análise.....	44
4.2.1. Nomenclatura de Equipamentos Médicos.	44
4.2.1.1 Nomenclatura das US	44

4.2.1.2. Nomenclatura de Equipamento Geral e Específico	45
4.2.2. Codificação de Equipamentos Médicos.....	45
4.2.2.1. Codificação de US	46
4.2.2.2. Codificação de Equipamentos	46
4.2.3. Construção do Modelo de Requisitos.....	47
4.2.3.1. Identificação de actores.....	47
4.2.3.2. Identificação de use cases para cada actor.....	48
4.2.3.3. Descrição de Use Cases	51
4.2.3.4. Especificação de Interface	58
4.2.4. Construção do modelo de análise	58
4.2.4.1. Use Case Inventariação.....	60
4.2.4.2. Use Case Folha de Obra.....	61
4.3. Projecto do Sistema	63
4.3.1. Construção do Modelo de Projecto.....	63
4.3.2. Modelo de Implementação.....	64
4.3.2.1. Arquitectura de implementação.....	67
4.3.2.2. Modelo de Teste.....	67
4.3.2.3. Informação Geral Sobre os Custos de Implementação	68
4.3.2.4. Benefício Organizacional após Implementação do Sistema	68
Capitulo V: Conclusões e Recomendações.....	70
Capitulo VI: Bibliografia.....	73
ANEXO A: Guião de Entrevista Estruturada	I
ANEXO B: Guião de Entrevista Semi-Estruturada	III
ANEXO C: Figuras Ilustrativas das Metodologias OO.....	IV
ANEXO D: Modelo de Teste.....	VII
D1. Teste Módulo – Inventariação de Equipamento	VII
D2. Teste Módulo – Abertura de FO	IX
D3. Teste Módulo – Fecho de FO.....	XI
D4. Teste Módulo – Impressão de Relatórios.....	XIII
ANEXO E: Ecrãs de Comparação do Sistema Legado e do Novo.....	XV

Índice de Figuras

Figura 2.1 Elementos sintácticos para a representação de Use Cases em OOSE.....	11
Figura 2.2 Elementos sintácticos do modelo de análise em OOSE.....	11
Figura 2.3 Elementos sintácticos do diagrama de interacção em OOSE.....	13
Figura 2.4 Fases de Modelação de sistemas em Booch. (Adaptado de Hawryszkiewicz, 1998).....	14
Figura 2.5 Fases de modelagem de sistemas em OMT.....	15
Figura 2.6 Fases de modelagem de sistema em Fusion.....	18
Figura 2.7 Interoperabilidade entre Sistemas Legados e Novos Sistemas.....	26
Figura 2.8 Processo de Migração de Sistemas de Informação Legados (Adaptado de Recchia et al)....	29
Figura 4.1 Descrição do Sistema.....	43
Figura 4.2 Diagrama de Use Cases do sistema em estudo.....	50
Figura 4.3 Relatórios a serem emitidos pelo sistema e seus destinos.....	54
Figura 4.4 Lay-out de Tela Principal do Sistema.....	58
Figura 4.5 Modelo de Objecto do Sistema em Estudo.....	59
Figura 4.6 Modelo de objectos referente a use case “Inventariação”.....	60
Figura 4.7 Modelo de objecto referente a Use Case “Folha de Obra”.....	61
Figura 4.8 Modelo de blocos do sistema.....	63
Figura 4.9 Diagrama de interacção para use case "Inventariação".....	64
Figura 4.10 Diagrama de interacção para use case "Folha de Obra".....	64
Figura 4.11 Modelo físico de base de dados do sistema em estudo.....	66
Figura 4.12 Arquitectura de Implementação do Sistema em Estudo.....	67
Figura C.1 Elementos Sintácticos de Análise e Desenho em FUSION.....	IV
Figura C. 2 Elementos Sintácticos de Análise e Desenho em OMT.....	V
Figura C.3 Elementos Sintácticos de Análise e Desenho em COAD-YOURDON.....	VI
Figura E.1 Formulário de Inventariação de Equipamento no Sistema Legado.....	XV
Figura E 2 Formulário de Inventariação de Equipamento no Novo Sistema.....	XV
Figura E.3 Formulário de Inventariação de Unidade Sanitária no Sistema Legado.....	XVI
Figura E.4 Formulário de Inventariação de Unidade Sanitária no Novo Sistema.....	XVI
Figura E.5 Formulário de Registo de Requisição de Intervenção.....	XVII
Figura E.6 Formulário de Abertura e Fecho F.O no Sistema Legado.....	XVIII
Figura E.7 Formulário de Abertura e Fecho F.O no Novo Sistema.....	XVIII

Figura E.8 Formulário de Registo de Funcionário no Sistema Legado.....	XIX
Figura E.9 Formulário de Registo de Funcionário no Novo Sistema.....	XIX

Índice de Tabelas

Tabela 2.1 Comparação na perspectiva estática das metodologias OO.....	22
Tabela 2.2 Comparação na perspectiva dinâmica das metodologias OO.....	23
Tabela 3.1 Sistemas de Informação Legados desenvolvidos e em uso SIS (Adaptado: MISAU 2003 e Nhampossa 2006)	34
Tabela 4.1 Lista de níveis de atendimento e administrativos no SNS. (Fonte: MISAU, 2003)	45
Tabela 4.2 Resumo de indicadores do Sistema de Manutenção. (Fonte: MISAU, 2003)	57

CAPITULO I

INTRODUÇÃO E OBJECTIVOS

Capítulo I: Introdução e Objectivos

1.1. Introdução

O processo de desenvolvimento de sistemas vem sofrendo uma crescente dinâmica onde a qualidade, o custo e o tempo são factores determinantes para o seu sucesso. Sendo assim, a utilização de técnicas consideradas fundamentais para análise e desenho de sistemas torna-se imprescindível. Tais técnicas permitem que durante o processo de desenvolvimento de sistemas sejam utilizados elementos cuja eficácia e eficiência foram previamente comprovadas. Dentre as técnicas, a padronização e a reutilização (Silva e Santander, 2000) podem ser citadas como duas das mais eficazes, pois permitem que os desenvolvedores de sistemas sigam critérios convencionais durante o ciclo de vida de desenvolvimento de sistemas.

Por padrões de software pode-se referir à diferentes níveis de abstracção no desenvolvimento de sistemas orientados a objectos. Assim, pode-se ter desde padrões de análise (nível mais alto) passando por padrões de desenho até padrões de código. Estes padrões, sejam eles de análise, de desenho ou de código, são facilmente alcançáveis com o uso das chamadas metodologias de desenvolvimento de sistemas como é o caso das metodologias orientadas a objectos.

O crescimento das organizações e a necessidade cada vez mais crescente de obtenção de informação útil faz com que estas optem desde cedo pela utilização de sistemas de informação desenvolvidos de maneira diferenciada e que, por vezes, não vão de acordo com a missão estratégica da organização. Nestes casos a tendência destas organizações é de construir ou adquirir no mercado novos sistemas com especificações mais próximas às da sua missão estratégica.

O desenvolvimento de sistemas a partir do “zero” é, na maioria das vezes, mais difícil que quando existe algo pronto para usar como ponto de partida. Por exemplo, em muitos sistemas de informação podem-se aproveitar a estrutura geral de um programa existente, os módulos para montagem de menus, os relatórios, as consultas e a estrutura de inserção, a alteração e a eliminação de dados. Esse aproveitamento pode ser de grande ou pequena envergadura, dependendo do tipo de aplicação a ser desenvolvida e de acordo com os sistemas disponíveis.

Ao implementar reformas nas organizações em geral e, particularmente, no sector de saúde e, especificamente, no Departamento de Manutenção (DM) do Ministério da Saúde (MISAU), deve-se ter especial atenção no papel dos Sistemas de Informação Legados (SIL) existentes. Normalmente, SIL são sistemas antigos e diversificados com código fonte fechado, documentação deficiente, diferentes formatos de armazenamento de dados e difícil partilha de informação entre estes sistemas. Tal situação sugere a transição destes sistemas para novos com uma plataforma e filosofia mais aberta e flexível.

O DM é um órgão do MISAU responsável pela gestão das tecnologias empregues no sector de saúde pública em Moçambique, concretamente a gestão de equipamento médico geral e específico, instalações técnicas especiais, edifícios e veículos. Neste departamento funciona um sistema de gestão de equipamentos médicos chamado SIM-ORGANIZER. Este sistema foi construído na plataforma Access 95, em 1996. Actualmente o SIM-ORGANIZER não responde às necessidades do departamento, sobretudo após a revisão da política de manutenção em 2003. Como consequência, o departamento enfrenta dificuldades administrativas e logísticas no controlo da manutenção de equipamentos sanitários. Por exemplo, utilizando este sistema não é possível obter o custo de manutenção de um determinado equipamento.

O presente trabalho apresenta uma das possíveis aproximações de soluções do problema do DM, nomeadamente a dificuldade crescente na gestão da manutenção de equipamentos médicos. A aproximação aqui apresentada passa pela migração do sistema de informação legado para um novo com recurso a metodologia de análise e de desenho Object Oriented Software Engineering (OOSE) e, ferramentas de migração Visual Basic .Net (VB.NET) e Sistema de Gestão de Base Dados (SGBD) MySQL.

Com este trabalho, espera-se uma avaliação da situação actual do sistema de informação (SI) do DM e identificação dos constrangimentos para o seu bom desempenho. Com a implementação dos ideais apresentados neste trabalho, espera-se uma melhoria no controlo de equipamentos existentes nas unidades sanitárias (US), conhecimento da localização de um equipamento e o seu estado funcional, quantificar as avarias e o custo de manutenção de um equipamento e impulsionar o colectivo do DM no uso das tecnologias de informação.

1.2. Motivação

A constante evolução tecnológica tem feito com que sistemas e aplicações evoluam e mudem de plataformas com uma frequência cada vez maior, tornando necessário a criação ou adaptação de uma arquitectura tecnicamente estável apoiada ao uso de ferramentas de modelação estandardizadas.

Face a esta dinâmica, as organizações vêm-se obrigadas a manter a sua missão estratégica e enquadrar-se neste ambiente dinâmico com exigências crescentes dos seus clientes. Ao analisar o seu SI, estas podem se deparar situações em que:

1. O SI não corresponde à sua missão estratégica;
2. O SI precisa de melhoria para se adaptar à evolução tecnológica;

Nos dois casos, a organização pode dentre várias outras optar por substituir completamente o seu SI ou explorar as características do actual SI, acrescentando-lhe as novas especificações.

Nas duas alternativas, surge uma questão: como fazer a substituição (ou melhoria) do SI?

É esta questão que constitui a base do presente trabalho de pesquisa. Com recurso a metodologias, técnicas ferramentas de referência, base teórica e empírica, apresenta-se uma abordagem à questão levantada. A fundamentação empírica foi encontrada no DM e este foi escolhido pelo facto de o autor ter trabalhado neste departamento como estagiário, e como tal percebido da necessidade de reestruturação do SI local.

1.3. Definição do Problema.

O crescimento do número de Unidades Sanitárias (US) no país e dos serviços nelas prestados aumenta também, a necessidade de aquisição de equipamentos médicos especializados de alta tecnologia. A necessidade de manter uma informação fiável acerca da localização desses equipamentos e o seu estado funcional aumentou no DM. Para além disso, é missão do DM monitorar os custos de manutenção do equipamento durante a sua vida útil.

Na tentativa de achar uma solução tecnológica para o fornecimento de informação fiável relativa a manutenção de equipamentos, o DM desenvolveu, em 1996, um sistema informático chamado SIM-ORGANIZER com o financiamento da Cooperação Italiana. O sistema foi desenvolvido por uma

equipa Italiana na plataforma Access 95 e funcionou até 2002. Com alteração da política de manutenção em 2003, o sistema ficou desactualizado (e não funcional), não se enquadrando na missão estratégica actual do DM.

Como não funcionamento do SIM-ORGANIZER o DM denota dificuldades no fornecimento de informação sobre os equipamentos relativamente à sua aquisição, localização, condições de funcionamento, consumíveis, acessórios, peças sobressalentes, movimentações, idade, número de avarias registadas durante a vida do equipamento, tempo de resposta dos serviços de manutenção à ocorrência de avarias, custo de manutenção ou reparação.

Aliados aos problemas acima descritos, acrescentam-se ainda a falta de procedimentos para a gestão de informação, a falta de avaliação dos processos de gestão de informação, a pouca qualidade de informação disponibilizada a nível central que é consequência da falta de capacidade de obtenção e utilização da informação disponível.

É tecnologicamente e administrativamente difícil adaptar o SIM-ORGANIZER às necessidades actuais do DM, pois o código fonte do mesmo não existe a nível do DM nem do MISAU; para além disso, a documentação deste sistema também não existe. Pretende-se, com este trabalho, fazer a migração do SIM-ORGANIZER com nova especificação de análise e desenho com auxílio de novas plataformas de transição sem, contudo, ignorar as especificações do mesmo de modo a produzir um sistema que responda às necessidades actuais da manutenção de equipamentos.

1.4. Objectivos

1.4.1. Objectivo Geral

- Desenvolver um modelo de migração de SIM-ORGANIZER adoptando metodologias e ferramentas orientados a objectos.

1.4.2. Objectivos Específicos

- Identificar, analisar e comparar as metodologias orientadas a objectos;
- Explorar a aplicabilidade das metodologias orientadas a objectos para migração de SIL;
- Identificar e analisar as diferentes ferramentas de migração de SIL;
- Analisar a infra-estrutura de sistemas de informação do DM;
- Conceber o modelo de migração do SIM-ORGANIZER, usando uma metodologia de análise e de desenho orientado a objectos;
- Testar e avaliar com os utilizadores o modelo de sistema de migração de modo a determinar se o modelo resolve os problemas previamente identificados.

Este relatório está dividido em seis capítulos a seguir descritos.

O primeiro capítulo é a introdução. Neste capítulo encontra-se a definição problema, a motivação e os objectivos geral e específicos. O segundo capítulo está reservado à revisão de literatura da qual se extraíram os conceitos base para a concretização deste trabalho. São abordadas neste capítulo matérias referentes às metodologias de análise e de desenho de sistemas orientados a objectos, Sistemas de Informação Legados e ferramentas de migração de Sistemas de Informação Legados. O terceiro capítulo descreve a metodologia de pesquisa usada para a realização deste trabalho. No quarto capítulo apresenta-se a modelação do sistema em estudo, no qual são aplicados os conceitos descritos no segundo capítulo para o desenvolvimento do modelo de sistema de migração. O quinto capítulo apresenta as conclusões e as recomendações seguido do sexto capítulo referente à bibliografia usada neste relatório. E, por último, encontram-se os anexos.

CAPITULO II

REVISÃO DE LITERATURA

Capítulo II: Revisão de Literatura

2.1. Metodologias Orientadas a Objectos

O desenvolvimento de sistemas requer a utilização de métodos e ferramentas que possibilitem avaliar se o sistema está sendo construído efectivamente seguindo padrões convencionais, daí o surgimento das Metodologias de Desenvolvimento de Sistemas.

O termo Metodologia apresenta várias definições, dependendo da perspectiva em análise. Na perspectiva de desenvolvimento de sistemas, Ricardo (1996) apresenta uma definição geral como “procedimentos sistemáticos de geração de modelos que descrevem aspectos de um sistema sob desenvolvimento”. Na mesma perspectiva, Hawryszkiewicz (1998) apresenta uma definição com um pouco mais de detalhes, “que consiste num conjunto de procedimentos, normas, técnicas e ferramentas que podem ser usadas durante o processo de desenvolvimento de sistemas”.

Olhando pelas definições acima apresentadas, nota-se a existência de um denominador comum: procedimentos para o desenvolvimento de sistemas. Constitui uma boa prática o uso de alguma metodologia durante o desenvolvimento de sistemas.

Segundo Hawryszkiewicz (1998), as metodologias indicam os procedimentos a seguir para a determinação dos requisitos de sistema e que permitem o desenvolvimento de sistemas de qualidade¹, bem como a documentação do sistema. Para além das vantagens apresentadas, Bennet et al (2002) acrescenta ainda mais vantagens:

- Fácil comunicação entre diferentes intervenientes no processo de desenvolvimento do sistema, uma vez que usam os mesmos princípios, regras apresentadas pela metodologia;
- Manter a continuidade do projecto de desenvolvimento caso ocorra a desistência de parte ou toda a equipa de desenvolvimento, pelo facto de as metodologias permitirem a documentação do sistema em desenvolvimento;
- Integração de novos membros na equipa;
- Elevar o conhecimento dos elementos envolvidos no processo de desenvolvimento.

¹ Sistema de qualidade é um sistema funcional e que satisfaz os requisitos do utilizador (Hawryszkiewicz, 1998)

A história regista dois grandes grupos de metodologias (Bennet et al, 2002):

- As metodologias estruturadas – modelo de dados relacional;
- As metodologias orientadas a objectos (OO) – modelo de dados OO.

As metodologias estruturadas propõem o desenvolvimento de sistemas com uma especificação estruturada de todos os processos existentes: as técnicas de armazenamento de dados e as interfaces com outros subsistemas (Bennet et al, 2002). O desenvolvimento de sistemas usando as metodologias estruturadas é baseado em: (Silva e Santander, 2000)

1. Definir os requisitos do sistema;
2. Identificar os eventos – estímulos, aos quais o sistema deve responder para atingir ao seu propósito;
3. Identificar os dados que devem ser armazenados e a actividade que deve ser executada para que o sistema responda completamente ao evento.

As técnicas e notações nestas metodologias são essencialmente os Diagramas de Fluxo de Dados (DFD), e os diagramas de entidade e relacionamentos.

Um exemplo típico deste grupo de metodologia é a Structured Systems Analysis and Design Method (SSADM).

As metodologias orientadas a objectos descrevem o sistema como um conjunto de objectos pertencentes a classes que interagem entre si (Hawryszkiewicz, 1998).

2.1.1. O Paradigma Orientação a Objectos

Paradigma é um conjunto de regras que estabelecem fronteiras e descrevem como resolver os problemas dentro destas fronteiras (Silva e Santander, 2000). O termo orientação a objectos significa organizar o mundo real como uma colecção de objectos que incorporam estrutura de dados e comportamento (Leite e Júnior, 2007). No paradigma de orientação a objectos, tudo pode ser potencialmente representado como um objecto. Para a percepção deste paradigma, os conceitos classe e objectos são fundamentais.

Uma **classe** é a descrição de um grupo de objectos com propriedades semelhantes (atributos), mesmo comportamento (métodos), mesmos relacionamentos com outros objectos (associações e agregações) e

mesma semântica (O'Neill e Nunes, 2001). Enquanto que o **objecto** denota uma entidade, seja ela de natureza física, conceitual ou de software do problema em estudo a respeito da qual se pretende manipular informação (O'Neill e Nunes, 2001). Em relação as classes e objectos associam-se os conceitos de:

- **Herança**, que pode ser entendido como sendo um conjunto de classes criadas a partir de uma ou mais classes com características semelhantes e os elementos desse subconjunto herdaram todas as características do conjunto original (Leite e Júnior, 2007);

Para exemplificar a ideia de Herança, consideremos uma classe *Equipamento* que tenha entre outros os atributos como *data de fabrico* e *origem*. Podemos construir uma outra classe *Veículo* que herda a classe *Equipamento*. Neste momento a classe *Veículo* terá os atributos *data de fabrico* e *origem* mas também poderá ter outros atributos como, por exemplo a *matricula*.

- **Encapsulamento** é a base de toda a abordagem do paradigma orientado a objectos; isto porque contribui fundamentalmente para diminuir os malefícios causados pela interferência externa sobre os dados (Leite e Júnior, 2007). Partindo desse princípio, toda e qualquer transacção feita com esses dados só pode ser feita da através de procedimentos (métodos) colocados “dentro” desse objecto;

Para a percepção do conceito “encapsulamento”, imagine um equipamento que tenha os botões *Ligar*, *Desligar*, etc. Estes botões executam uma série de operações existentes no equipamento, onde são executadas pelos componentes existentes dentro do equipamento. Não interessa ao médico (utilizador) saber como é o funcionamento interno do equipamento; esta informação só é relevante para os projectistas do equipamento. As informações pertinentes ao usuário do equipamento são as existentes no meio externo (botões) que activam as operações internas do equipamento.

- **Polimorfismo** esta propriedade no paradigma orientado permite que objectos de diferentes classes possam receber um mesmo formato de mensagem.

Um exemplo de aplicação do polimorfismo: consideremos que na linguagem VB.NET temos as classes *Listview* e *ComboBox*; cada uma destas classes tem o método *clear* que serve para limpar os itens em cada uma delas.

2.1.2. Object Oriented Software Engineering (OOSE)

2.1.2.1. Aspectos Gerais de OOSE

A metodologia OOSE, desenvolvida por Jacobson e Outros e lançada em 1992, é uma metodologia orientada a objectos cuja abordagem é classificada como dirigida ao cenário.

Nas metodologias dirigidas ao cenário, são identificadas, inicialmente, todas as situações de processamento a que o sistema pode ser submetido e são descritos os possíveis diálogos entre o sistema e o meio externo.

O desenvolvimento de um sistema em OOSE passa pelas etapas de Análise e Construção. Na etapa de Análise é construído o *Modelo de Requisitos* e o de *Análise* e, na etapa de Construção, é desenvolvido o *Modelo de Projecto* e o de *Implementação* (Jacobson et al, 1992)

O principal elemento de descrição da metodologia são os use cases², que em português significa Casos de Uso. *Use Case* descrevem os requisitos funcionais do sistema em OOSE; descrevem, também, como os utilizadores interagem com o sistema e que situações de processamento o sistema deve responder. Os Use Case providenciam um mecanismo de comunicação entre os *Stakcholders*³ (Papajorgji e Pardalos, 2006). As várias etapas de descrição de um sistema em OOSE baseiam-se na definição de use cases.

2.1.3.2. Técnicas de Modelagem para a Descrição de Sistema em OOSE

A metodologia OOSE utiliza um conjunto de modelos para descrever um sistema: o *modelo de requisitos*, o *modelo de análise*, o *modelo de projecto*, o *modelo de implementação* e o *modelo de Teste* - sendo que estes dois últimos produzem o sistema pronto para a utilização.

O modelo de requisitos possui como principal elemento o diagrama de use cases, onde são representados actores, use cases e ligações entre eles – sem detalhe dos elementos, mas apenas os seus nomes. O restante da descrição de use cases é feito de forma textual. Sua finalidade é descrever as

² O termo Use Case é mantido na sua versão em Inglês por ser característica da metodologia.

³ Em desenvolvimento de sistemas de Informação Stakeholders são utilizadores, clientes, equipe de desenvolvimento e gestores.

situações de processamento a que o sistema pode ser submetido, sem descrevê-lo internamente, mas apenas numa visão “caixa preta” do diálogo entre os actores e o sistema.

O modelo de análise é um conjunto de modelos de objectos compostos por três tipos de classes:

- As classes entidade – que são as classes do domínio do problema já identificadas;
- As classes de interface – responsáveis pela comunicação entre o sistema e o meio externo;
- As classes de controle – que concentram funcionalidades específicas de determinados use case.

Este modelo é complementado pela descrição textual associada aos diagramas (descrição das Classes e dos Use Cases).

A construção do modelo de objectos do sistema é feita a partir de vários diagramas de objectos, cada um descrevendo um use case. A funcionalidade de cada classe está distribuída em vários diagramas e a construção de cada diagrama consiste em identificar que classes participam do use case (classes entidade, de interface e de controle) e como estão associadas.

O modelo de projecto inclui tanto a descrição estática e dinâmica do sistema como um todo, quanto das classes, em particular. A estrutura de blocos reflecte a mesma estrutura estática definida na análise - excepto pelas alterações que ocorrem no projecto. Os diagramas de interacção produzem uma descrição dinâmica do sistema (composto por um conjunto de blocos). A descrição da interface dos blocos consiste em relacionar os seus atributos e a assinatura de seus métodos na sintaxe da linguagem de programação adoptada para a implementação.

A última actividade de projecto consiste em produzir a descrição do corpo dos blocos, ou seja, em definir os algoritmos dos métodos. Isto é feito no projecto do comportamento dos módulos, que utiliza diagrama Specification Description Language (SDL) – Linguagem de Especificação e Desenho.

2.1.2.3. Elementos Sintácticos das Técnicas de Modelagem em OOSE

Modelo de Requisitos

Define a fronteira de sistema e seu comportamento; consiste num modelo de use case, descrições de interfaces e especificação do domínio do problema. O modelo de use case é descrito com recurso a actores e use case (Dori, 2002).

A Figura 2.1 apresenta a representação de use cases do modelo de requisitos de OOSE. Ícones representam actores e elipses os use cases. Os use cases estão contidos em um rectângulo que representa o limite do sistema. Os actores recebem identificadores que descrevem o papel por eles representado no sistema. As elipses que representam use cases são rotuladas com seus identificadores.

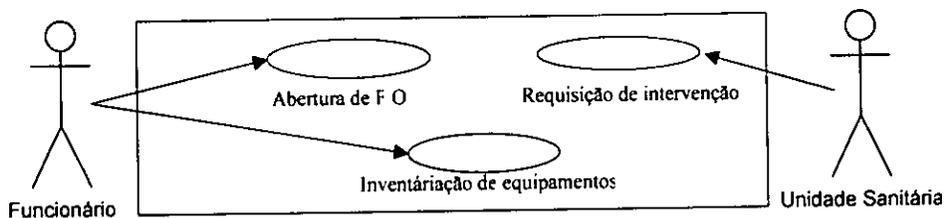


Figura 2.1 Elementos sintáticos para a representação de Use Cases em OOSE

Modelo de Análise

Nesta fase faz-se a estruturação do sistema sem ter em conta os detalhes de implementação. O objectivo é a captura de informação, comportamento e apresentação do sistema. O modelo de dados usado nesta fase é o modelo objecto entidade (Dori, 2002).

Os elementos sintáticos para a construção de diagramas do modelo de análise estão apresentados na Figura 2.2.

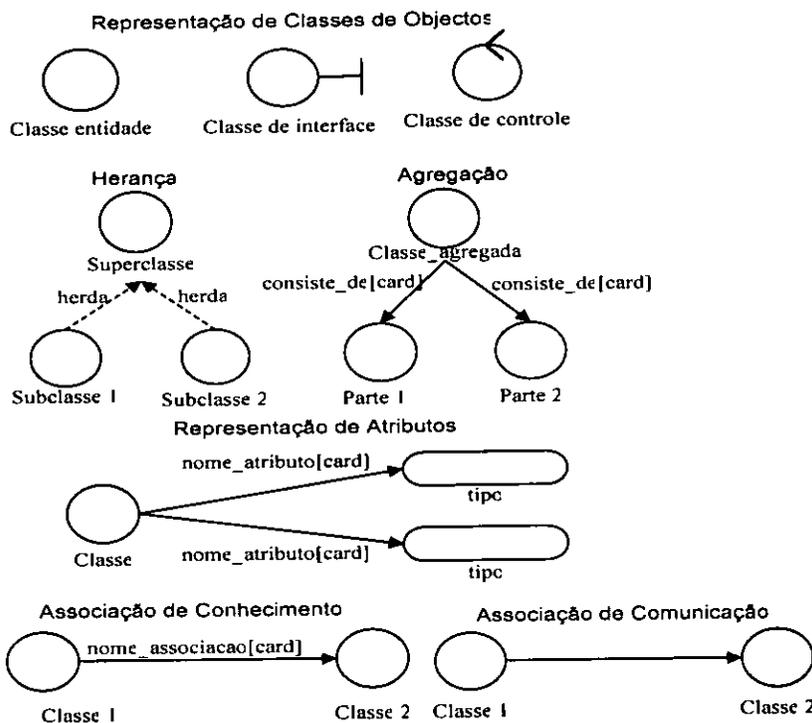


Figura 2.2 Elementos sintáticos do modelo de análise em OOSE

As classes de objectos têm formato circular e são diferenciadas conforme os tipos definidos em OOSE. Os atributos são representados por rectângulos de cantos arredondados conectados ao símbolo de classe. A representação de atributo inclui nome do atributo, tipo e cardinalidade. A técnica de modelagem não comporta a inclusão de métodos.

OOSE classifica associações em dois tipos: estáticas e dinâmicas. O grupo das associações estáticas inclui herança, agregação e associação de conhecimento - sendo esta uma associação entre instâncias em que uma instância conhece a existência de outra instância (manutenção de referência). A associação de conhecimento é unidireccional - uma seta aponta para a classe conhecida. Sua representação inclui cardinalidade. Esta associação se reflectir-se-á em atributos na implementação. A associação bidireccional demanda duas associações de conhecimento; esta não capacita o objecto a trocar informações com outro (Ricardo, 1996)

Associações dinâmicas são as associações de comunicação. Através desta associação, um objecto envia e recebe estímulos, o que corresponde a ocorrência de eventos. A representação da associação de comunicação é feita a partir de uma seta indicando quem procede o envio. A seta não é rotulada, assim, o significado da comunicação representada no diagrama deve ser expresso na descrição textual (a ausência de rótulo diferencia sintacticamente esta associação da associação de conhecimento). A associação de comunicação pode representar o envio de uma mensagem para a execução de um método ou abstrair uma interacção mais complexa que envolva várias trocas de mensagens (identificando pela seta quem toma a iniciativa da interacção).

Modelo de Projecto

A estrutura estática do modelo de projecto consiste em um modelo de objectos de sintaxe idêntica do modelo de análise, excepto as classes que são chamadas de blocos e representadas por rectângulos. OOSE admite o uso de apenas um diagrama ou de vários por use case, como na análise.

No diagrama de interacção de OOSE, as classes de objectos são representadas por traços verticais, sendo os eventos (estímulos) por setas e a ordem de eventos se dá-se de cima para baixo (vide a Figura 2.3). Neste tipo de diagrama, uma barra vertical diferenciada representa a borda do sistema (destacando os estímulos que entram e saem).

Os estímulos podem ser diferenciados entre mensagens (estímulos intra-processos) e sinais (estímulos inter-processos). Mais de um traço vertical pode ser associado a uma classe, representando suas instâncias; a duração da execução de uma operação é destacada por um rectângulo sobre o traço do respectivo objecto; e pseudocódigo (texto estruturado), colocado à esquerda do diagrama permite a representação de execução condicional.

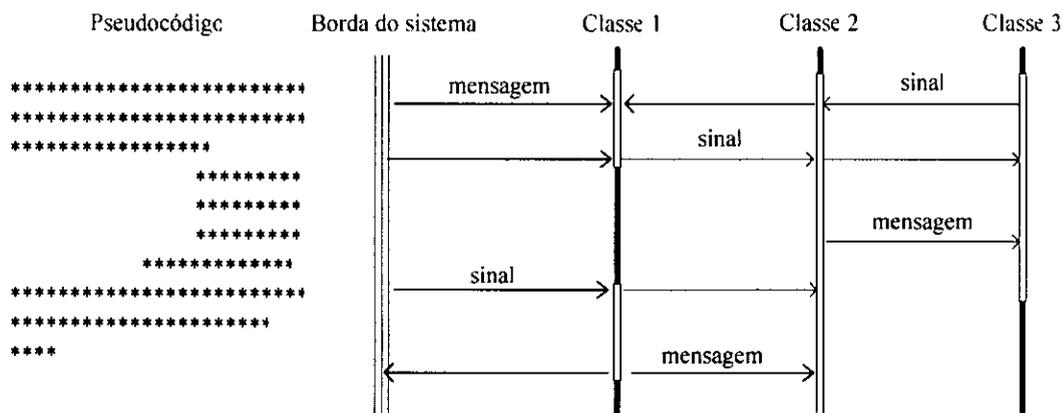


Figura 2.3 Elementos sintáticos do diagrama de interacção em OOSE

Para a descrição do comportamento interno dos blocos (informação necessária para o desenvolvimento de algoritmos de seus métodos) é utilizada a notação SDL.

Modelo de Implementação

Não tem nenhum diagrama que descreva este modelo; faz a unificação dos modelos de requisito, de análise e de projectos convertendo-os às características da linguagem de programação escolhida.

Modelo de Teste

Permitirá testar o sistema através de especificações de teste derivadas dos use cases.

2.1.3. Grady Booch

A metodologia Grady Booch é uma metodologia orientada a objectos desenvolvida por Grady Booch cuja a abordagem é dirigida a dado. Nas metodologias dirigidas a dado adoptam como primeiro procedimento a identificação de classes de objectos que compõem o sistema (Ricardo, 1996). O ciclo de vida de desenvolvimento de sistemas, usando esta metodologia passa por quatro (4) fases, a Figura 2.4 ilustra estas fases.

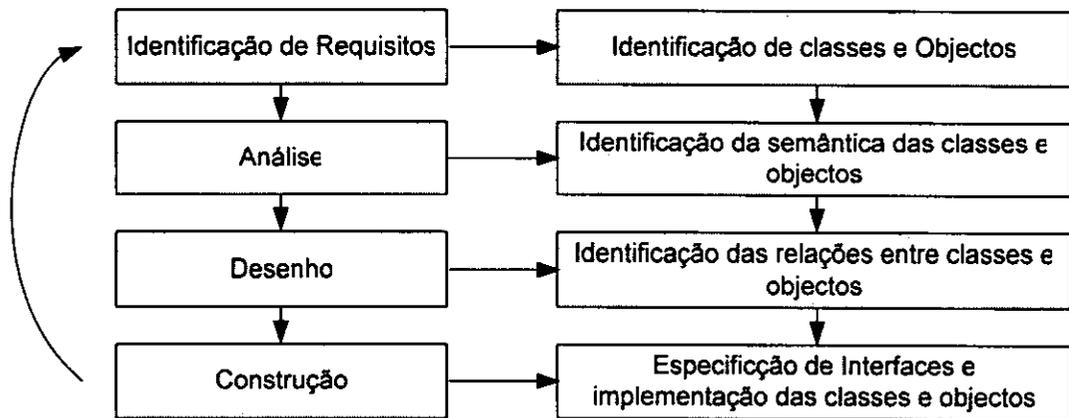


Figura 2.4 Fases de Modelação de sistemas em Booch. (Adaptado de Hawryskiewicz, 1998)

Como mostra a Figura 2.4, a primeira fase de desenvolvimento de sistemas em Booch é a identificação dos requisitos do sistema, que consiste na identificação de abstrações das classes e objectos que fazem parte do sistema, estabelecimento dos limites do problema e obtenção da primeira decomposição do problema. A ênfase, nesta fase, é dada à obtenção de abstrações mais significativas do domínio do problema.

O resultado desta fase é um dicionário de dados que contem a descrição precisa das classes e objectos identificados.

A fase seguinte é a de análise, cuja principal actividade é a identificação da semântica das classes e objectos, que consiste em determinar o comportamentos, atributos, e responsabilidades.

O resultado desta fase é um conjunto completo de responsabilidades e operações para cada classe e objecto identificados e são apresentados sob forma de diagramas de interacção e diagrama de objectos.

Na fase de desenho são identificadas as relações entre as classes e objectos; as classes e objectos com características semelhantes são agrupadas para determinar as heranças e polimorfismo. Os resultados desta fase são apresentados sob forma de diagramas de classes e diagramas de colaboração.

A última fase é a de construção. Nesta fase, as classes e objectos são refinadas até um nível mais detalhado; novas classes são adicionadas, se necessário, são especificadas as interfaces, são seleccionados os algoritmos e as classes e objectos são implementadas na linguagem de programação seleccionada.

A característica mais notável da metodologia Grady Booch é o conjunto de notações mais expressivas: usa seis tipos de diagramas (diagrama de objectos, classes, interacção, transição de estados, módulos e colaboração) cada um com um vocabulário bastante elaborado que permite que sejam modelados muitos aspectos do sistema em estudo.

2.1.4. Object Modeling Techique – Técnica de Modelação de Objectos (OMT)

A Metodologia OMT foi originalmente proposta por Rumbangh e Outros em 1991, contudo, a mesma foi revista em 1996 por Rumbangh (Hawryszkiewicz, 1998). Esta metodologia quanto à abordagem é dirigida a dado. As metodologias dirigidas a dado adoptam, como primeiro procedimento, a identificação de classes de objectos. Esta metodologia é considerada por vários autores como sendo metodologia para teste de sistemas, pois considera a fase de análise de requisitos como já existente (Hawryszkiewicz, 1998). A construção de sistemas usando esta metodologia passa por três (3) fases: Análise, Desenho e Construção. Cada uma destas fases produz um modelo complementar que expressa os aspectos do problema e as suas soluções. Esta metodologia enfatiza o processo iterativo, isto é, a informação obtida numa determinada fase do processo complementa ou até podem alterar os resultados das etapas anteriores. A Figura 2.5 apresenta as três fases de modelagem de sistemas em OMT.

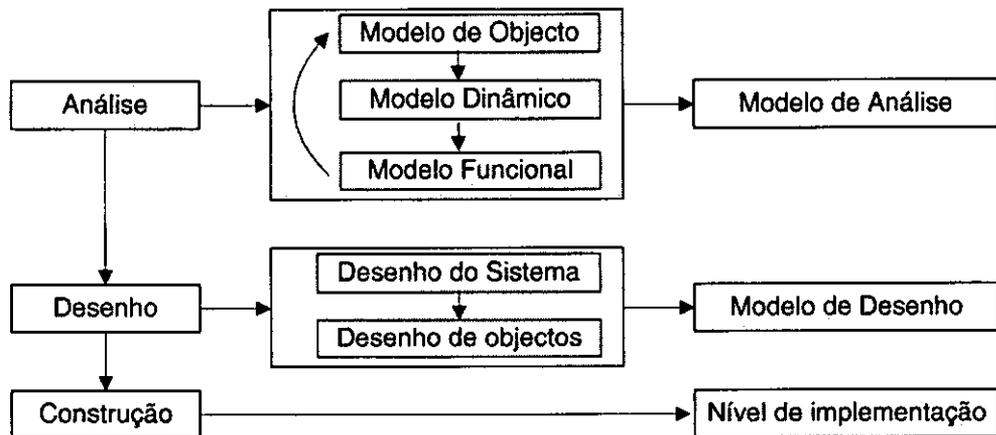


Figura 2.5 Fases de modelagem de sistemas em OMT

2.1.4.1. Fase de Análise

A metodologia OMT concentra a sua atenção nesta fase, que consiste numa primeira abstracção da descrição da informação inicial do problema, seguida de construção do modelo de objectos, modelo dinâmico e o modelo funcional.

Modelo de Objectos

Este modelo caracteriza a estrutura estática de objectos do sistema, olhando pelas diferenças e semelhanças dos objectos análogos, formando classes e identificando as relações entre as classes. O modelo de objectos é expresso em um ou mais diagramas de objectos representados como rectângulos contendo o nome da classe, os seus atributos, os métodos e o relacionamento entre as classes.

Modelo Dinâmico

Descreve as interacções sequenciais e temporais entre objectos do sistema, isto é, os estímulos que ocorrem e a resposta destes estímulos por parte dos objectos. O modelo dinâmico é apresentado através do diagrama de estado composto por estados e transições de estado através de ocorrência de um determinado evento.

Modelo Funcional

Define as situações de processamento a que o sistema pode ser submetido, determina os mecanismos de como as variáveis de entrada são processadas para a obtenção das variáveis de saída. O modelo funcional é expresso através de DFD composto de processos, actores externos, repositório de dados e fluxo de dados interligando estes elementos.

O resultado da fase de análise é o modelo completo de análise.

2.1.4.2. Fase de Desenho

Nesta fase, a equipa de desenvolvimento adopta uma estratégia de alto nível para a arquitectura do sistema; são estabelecidas as regras de negocio que servirão como base para as fases subsequentes. É nesta fase em que faz-se a especificação de interface do sistema em estudo. O desenho de sistema e o desenho de objectos são as actividades fundamentais desta fase.

Desenho de Sistema

Faz-se a escolha da estratégia básica para a implementação dos repositórios de dados em termos de estrutura de dados, arquivos e bancos de dados; identificam-se os recursos globais e determinam-se mecanismos para o controlo do acesso a dados.

Desenho de Objectos

Nesta fase, entre outras actividades, faz-se o agrupamento das classes e associações em módulos; obtém-se os métodos para o modelo de objectos, faz-se a escolha dos algoritmos para a implementação de métodos. O resultado da fase de desenho é o modelo completo de desenho.

2.1.4.3. Fase de Construção

É a transformação dos modelos descritos nas fases de análise e de desenho numa linguagem de programação escolhida, que pode ser uma orientada a objecto como não. O resultado da fase de construção é o sistema construído pronto para implementação.

2.1.5. Fusion

A metodologia Fusion que foi desenvolvida por uma equipa vinculada à HP (Hewlett-Packard) lançada em 1992, é uma metodologia orientada a objectos, com uma abordagem dirigida a dado. O desenvolvimento de um sistema em Fusion passa pelas etapas de *análise, desenho e implementação*.

A fase de análise consiste em determinar os objectos do sistema e as suas interacções através da construção de dois modelos, nomeadamente o *modelo de objectos* e o *modelo de interface* (vide a figura 2.6).

A estrutura estática (chamada de modelo de objectos em Fusion) é mostrada em um diagrama de classes construído seguindo a ideia de entidades e relacionamento semelhante a da OMT. O modelo de interface contém a modelagem dinâmica desenvolvida na etapa de análise.

Durante a análise, não há preocupação com os métodos das classes, mas sim com as funcionalidades do sistema. São identificadas as possíveis interacções do sistema a desenvolver com o meio externo. A fase de análise é um processo iterativo e o sistema é sempre visto como um todo e não como um conjunto de objectos interagindo. Esta de análise termina quando os modelos estiverem completos e consistentes.

A fase de desenho resulta da fusão de quatro modelos (denominados grafos) que são derivadas essencialmente na ordem indicada na Figura 2.6.

O grafo de interacção – assemelha-se aos diagramas de colaboração em OMT; este modelo descreve como os objectos do sistema interagem entre si em tempo de execução do sistema: que objectos são envolvidos em uma computação e como eles são combinados para a execução de uma determinada especificação.

O grafo de visibilidade – neste modelo descreve-se como são percebidas as comunicações entre objectos, identificando-se para cada objecto quais os outros que devem ser referenciados e como são feitas essas referencias.

O grafo de descrição de classes – na fase de análise não se dá ênfase à descrição das classes de objectos; é no grafo de descrição de classes que se estabelecem as operações e o conjunto de atributos para cada classe identificada.

Finalmente, **o grafo de heranças** – é um diagrama de classes que mostra as relações de herança das classes, as descrições das classes; são actualizadas para reflectir o desenho final do sistema.

A implementação consiste na tradução das especificações produzida, em uma linguagem de programação orientada a objectos.

Um dicionário de dados é construído ao longo das fases de análise e de desenho, descrevendo de forma textual as classes, os atributos, os métodos e os conceitos presentes na modelagem.

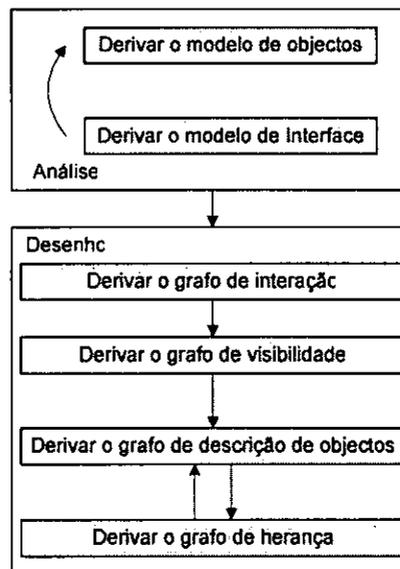


Figura 2.6 Fases de modelagem de sistema em Fusion

2.1.6. Coad-Yourdon

“Um sistema construído usando uma metodologia orientada a objectos é aquele cujos componentes são partes encapsuladas de dados e funções, que podem herdar atributos e comportamentos de outros componentes da mesma natureza, e cujos componentes comunicam-se entre si por meio de mensagens” (Yourdon, 1991).

A metodologia Coad-Yourdon que foi desenvolvida por Peter Coad e Edward Yourdon em 1991, é uma metodologia orientada a objectos cuja abordagem é dirigida a dado semelhante da metodologia Fusion.

O processo de desenvolvimento de sistema em Coad-Yourdon passa pelas etapas de *análise, projecto e implementação*.

Na fase de análise (denominada OOA – OO Analysis), a metodologia concentra-se em especificar o domínio do problema (sem se preocupar com a solução), os requisitos do sistema e as classes fundamentais do domínio do problema. Esta é a fase fundamental da metodologia cujas actividades estão abaixo descritas:

Identificação de objectos – consiste em identificar entidades do domínio de problemas que serão responsabilidades do sistema, resultando num modelo de objectos e especificação “classe-&-objecto” que serve como descrição do sistema.

Identificação de estruturas – identificação de relações entre objectos identificados de modo a organizá-las em hierarquias e estruturas de dados.

Identificação de assuntos – identificação de agrupamentos de objectos identificados e das correspondentes hierarquias e estruturas, de modo a separá-los em grupos denominados assuntos.

Definição de atributos – especificação dos objectos identificados e definição de serviços, onde serão definidas as operações passíveis de serem executadas pelos objectos identificados, correspondendo as funcionalidades do sistema.

Na fase de projecto (denominada OOD – OO Design), identificam-se as classes do domínio da solução e implementação computacional em três componentes: (1) Componente de interacção com os usuários, que são classes responsáveis pela interface do sistema. (2) Componentes de estrutura e administração

de dados, que são classes responsáveis pela gestão, acesso, actualização dos elementos armazenados na fonte de dados e (3) Componentes de administração de tarefas, que são classes de responsáveis por monitorar as actividades dos demais componentes em tempo de execução e as regras de negócios que o sistema terá de corresponder.

Em cada modelo produzido, tanto na fase de análise como na fase de projecto, as especificações são acompanhadas pela mistura da descrição textual com modelos gráficos.

A fase de implementação consiste na codificação das especificações produzidas nas fases de análise e projectos numa linguagem orientado a objecto como não.

2.1.7. UML

A UML (Unified Modeling Language) é uma linguagem de especificação e desenho que combina técnicas das metodologias OOSE de Jacobson, Booch de Grady Booch e OMT de Rumbangh (Hawryszkiewicz, 1998) de modo a providenciar uma notação padrão para construir, visualizar e documentar sistemas orientados a objectos (O'Neil e Nunes, 2001).

Segundo Hawryszkiewicz, (1998), os componentes fundamentais da modelação de sistemas em UML são os seguintes:

Os diagramas de classes – seguem as notações usadas em OMT e também incorporam técnicas usadas na metodologia Booch e de outros métodos de modelação das estruturas orientadas a objectos.

No modelo de requisitos de UML são usados os diagramas de use case de OOSE, os diagramas de interacção em OOSE são chamados diagramas de cenário em UML

A UML usa também os diagramas de estado que descrevem as mudanças de estado de objectos e diagramas de implementação.

A UML é uma linguagem de especificação OO e não uma metodologia OO. Para usar a UML com sucesso é necessário adoptar algum tipo de metodologia de desenvolvimento de sistemas, especialmente em sistemas de grande porte onde a organização de tarefas é essencial (Fumo, 2001).

2.1.8. Análise comparativa das metodologias orientadas a objectos

A comparação das metodologia anteriormente descritas tem sido difícil porque ambas têm o mesmo propósito, análise e desenho orientados a objectos e a escolha de cada uma delas num projecto de desenvolvimento de sistema depende das características do sistema a ser desenvolvido e do domínio da metodologia por parte da equipa envolvida.

A modelagem de sistema, seguindo uma abordagem de uma metodologia orientada a objectos, deve ter a capacidade de descrever o sistema sob uma perspectiva estática e dinâmica. A perspectiva estática consiste na identificação e descrição dos objectos (classes), seus atributos e relacionamentos. Em contrapartida, a perspectiva dinâmica consiste na identificação e descrição da colaboração entre os objectos (negócio) e os métodos das classes.

Comparando a perspectiva estática das metodologias OO, Coad e Youdon agrupa o nome da classe e a relação de atributos e métodos, diferencia as classes abstractas e as classes concretas a partir da notação de “classe & objecto”. A representação de OMT é semelhante, porém, não diferencia classes abstractas; ela inclui tipagem para os atributos e relação de argumentos e tipo de retorno para os métodos. Em OOSE é possível incluir o nome da classe e a relação de atributos (com tipo e cardinalidade – esta última, característica exclusiva de OOSE); não há diferenciação de classes abstractas nem a relação de métodos. Em Fusion, classes abstractas são diferenciadas (escrevendo o nome da classe em itálico) e o símbolo da classe agrupa a relação de atributos, com tipagem.

Algumas destas metodologias tem características intrínsecas. A diferenciação entre as classes abstractas e as classes concretas de Coad e Yourdon é de mais fácil leitura que a de Fusion – as demais metodologias não tratam esta diferenciação. Em termos de informação associada à relação de atributos, há destaque para OOSE, que além da tipagem, inclui cardinalidade. E em termos de informação associada à relação de métodos, há destaque para OMT, que inclui relação de argumentos (com tipagem) e tipo de retorno. A Tabela 2.1 apresenta o resumo da comparação das metodologias na perspectiva estática.

Informação / Metodologia	Atributos	Métodos	Diferenciação de classes abstracta/concreta
Coad e Yourdon	Nome	Nome, argumentos	Sim
OMT	Nome, tipo	Nome, argumentos, retorno	Não representa
OOSE	Nome, tipo, cardinalidade	Não representa	Não representa
Fusion	Nome, tipo	Não representa	Sim

Tabela 2.1 Comparação na perspectiva estática das metodologias OO

Na perspectiva dinâmica, na metodologia de Coad e Yourdon, resume-se às associações de comunicação (dinâmicas) incluídas no modelo de objectos. A modelagem dinâmica das classes, isoladamente, é composta por um diagrama de transição de estado (para a classe) e um fluxograma (para cada método).

OMT utiliza os diagramas de eventos (como ferramenta de identificação do comportamento dinâmico) e os diagramas de fluxo de eventos (parte do modelo dinâmico). Além deste, o DFD, que constitui o modelo funcional, também descreve o comportamento do sistema. A descrição do aspecto dinâmico em OMT é mais expressiva do que a de Coad e Yourdon. A modelagem dinâmica das classes utiliza, isoladamente, statecharts, que são mais expressivos do que os diagramas de transição de estado usado por Coad e Yourdon.

OOSE inclui as associações dinâmicas do modelo de objectos (específicas por Use Case ou não) e os diagramas de interacção (semelhantes aos diagramas de eventos de OMT, porém, mais expressivos). A modelagem dinâmica das classes utiliza, isoladamente, o diagrama SDL, que corresponde a um diagrama de transição de estado com elementos sintácticos de fluxograma, o que permite a descrição do algoritmo dos métodos.

Fusion utiliza o modelo de interface (associação do modelo de operação e do modelo de ciclo de vida) para a modelagem dinâmica do sistema. A interacção do sistema com o meio externo é semelhante à abordagem do modelo de requisitos de OOSE, porém mais formalizado. A Tabela 2.2 apresenta o resumo da comparação das metodologias OO na perspectiva dinâmica.

Informação \ Metodologia	Aspectos de modelagem dinâmica incluídos no modelo de objectos	Enumeração das interacções em tempo de execução (colaboração)	Evolução de estados a partir da ocorrência de eventos
Coad e Yourdon	Conexões de mensagem	não representa	não representa
OMT	não representa	Diagrama de fluxo de eventos	Diagrama de eventos (cenários)
OOSE	Conexões dinâmicas	não representa	Diagrama de interacção (cenário)
Fusion	não representa	não representa	Grafo de interacção de objectos

Tabela 2.2 Comparação na perspectiva dinâmica das metodologias OO

2.1.10. Razões para Escolher a Metodologia OOSE

Depois de identificadas e descritas as especificações das diferentes metodologias de análise e desenho orientadas a objectos, foi escolhida para este trabalho a metodologia OOSE. A razão para sua escolha teve como base a simplicidade da metodologia, pois possui recomendações de como o processo deve ser desenvolvido, dando ênfase ao cenário como um todo. Também contribui o facto de esta metodologia ser bastante referenciada por outros autores da área de modelação OO.

Pelo facto de o Cenário, Use Cases e Actores serem elementos descritivos da metodologia OOSE, o mesmo adequa-se às características do sistema de gestão de equipamentos em estudo, visto que o cenário e a comunicação com as entidades fora do DM constitui a regra de negócio fundamental daquele departamento.

Para além dos pontos anteriormente apresentados, esta metodologia, mesmo sendo muito referenciada na modelagem OO constitui um “mostro adormecido” a nível do Departamento de Matemática e Informática (DMI) porque pouco ou nada é ensinado acerca desta metodologia. Isto constituiu um incentivo para o autor apresentar o estudo teórico e a sua aplicação prática da metodologia OOSE que é descrita como uma metodologia forte de análise e de desenho OO.

2.2: Sistemas de Informação Legados

Neste capítulo, faz-se a apresentação dos conceitos relacionados aos SIL, sua natureza, os problemas típicos de SIL e, no final apresenta-se as estratégias (ferramentas) usadas no processo de migração/transição destes sistemas para os novos.

2.2.1. Perspectiva de Sistemas de Informação Legado

Quando uma organização pretende mudar o seu sistema de informação, encontra problemas relacionados com a existência de uma base complexa instalada de software, os chamados SIL. Estes softwares fazem parte da história da existência e funcionamento da organização cujo seu abandono ou reestruturação deve ser feito gradualmente e quando necessário. Segundo Henderson (2000) “SIL são sistemas computadorizados constituídos por software, hardware, bases de dados, redes de comunicação e regras de negócios da organização”.

Os SIL são sistemas geralmente lentos, inflexíveis e com um grau de incerteza elevado para dar resposta às novas e competitivas tarefas da organização devido (por vezes) à sua antiguidade e até às plataformas em que estes sistemas correm. Contudo, os SIL desempenham um papel não menos importante nas organizações, por isso a sua substituição é uma estratégia organizacional complexa e, por vezes, arriscada devido a três (3) razões (Henderson, 2000):

1. Deficiente especificação do SIL, como por exemplo a documentação pode não existir ou pode ter sido perdida facto, que torna difícil especificar no novo sistema integrando todas as funcionalidades do sistema legado.
2. SIL são projectados com as regras de negócios bem definidas para evitar a fragilidade organizacional e nelas são embutidas as medidas de segurança específicas; se o sistema, é substituído essas regras também terão de mudar acarretando custos imprevisíveis e outras consequências.
3. Podem ter sido embutidos no SIL processos organizacionais cruciais e podem não ter sido documentados estes processos, substituindo o SIL pode conduzir a não implementação destes processos no novo sistema, factor este que contribui para à resistência a mudança por parte dos gestores organizacionais.

A necessidade de lidar com SIL num ambiente de negócio no contexto actual é deveras vezes crítico, pois, com o passar de tempo, as organizações tendem a mudar, tanto a sua estrutura como o seu posicionamento no seu ambiente. Estas mudanças são devido aos factores internos ou externos da organização, as leis do ambiente em que está inserida, mudanças administrativas e até a necessidade de redefinição da sua estrutura de informação, exigindo, deste modo, a modificação e/ou novos sistemas de informação, sendo neste contexto, inevitável que os SIL também mudem.

A mudança de SIL pode ser feita na óptica de Nhampossa (2006) em três perspectivas distintas: Embutir, Migração e Redesenvolvimento.

- Embutir – é incorporar o SIL existente num novo sistema;
- Migração – é manter as funcionalidade do SIL existente e migra-lo numa nova plataforma, acrescentando-lhe novas funcionalidades;
- Redesenvolvimento – é o desenvolvimento do SIL com uma nova arquitectura, sem contudo ignorar as especificações do legado.

2.2.2. Processo de Transição em SIL

O processo de transição de SIL deve ser feito depois de uma avaliação exaustiva da qualidade de sistema e quando haver mudanças significativas administrativas e as regras de negócio da organização (Rodriguez e Ferrante, 2001). Na nova arquitectura de informação não pode ignorar todos os sistemas herdados da organização, assim como o património de informação da organização. Uma das primeiras preocupações da nova arquitectura é de como tirar vantagens dos sistemas já implementados e em uso corrente. A interoperabilidade entre os sistemas legados e os novos sistemas, conforme mostra a Figura 2.7 deve ser o principal compromisso (Rodriguez e Ferrante, 2001).

Silva e Santander (2000) relatam, em seu artigo, que os sistemas legados possuem lógica de programação, decisões de projecto, requisitos de usuário e regras de negócio, que podem ser recuperados, facilitando sua reconstrução, sem perda da semântica.

A seguir, descreve-se as directrizes metodológicas apontadas por estes autores como decisivas para realizar a transição, visando melhorar/actualizar os sistemas legados:



Figura 2.7 Interoperabilidade entre Sistemas Legados e Novos Sistemas

2.2.2.1. Formação e treinamento da Equipa

O treinamento requer que a equipa que irá realizar a actualização/melhoramento do SIL tenha algumas qualidades que se julgam essenciais:

1. Conhecimento de novas metodologias (actualmente com grande enfoque nas metodologias orientadas a objectos);
2. Conhecimento de ferramentas CASE (Computer Aided Software Engineering) que facilitam o trabalho de migração;
3. Excelente ambiente de trabalho, não só em relação ao grau de relacionamento da equipa como também em relação ao local apropriado para o desenvolvimento do sistema.

É importante referir que estas qualidades são fundamentais para que o processo de migração seja mais efectivo e produtivo.

2.2.2.2. Organização da Base de Informações

A organização da base de informações é decisiva para facilitar a reengenharia de um sistema legado utilizando outra metodologia. As seguintes situações podem ser observadas:

1. Um sistema sem documentação – neste caso, normalmente estão disponíveis apenas as informações do código fonte do sistema;
2. Um sistema com documentação actualizada – neste caso, se temos uma documentação em ordem, o processo é facilitado, uma vez que podemos verificar a análise efectuada anteriormente e as actualizações que se tornaram necessárias com o passar dos anos. Temos um histórico completo do sistema, simplificando a organização;

3. Um sistema com uma documentação não actualizada – nesta situação (bastante comum), as equipas mal tiveram tempo de realizar as actualizações nos códigos fontes do sistema. Normalmente, as equipas que trabalham no sistema não foram as mesmas que o desenvolveram.

Em qualquer uma das situações acima descritas, devemos ter em mente que a equipa que irá realizar a tarefa de evolução de sistema legado, utilizando uma nova metodologia, não será a mesma que o desenvolveu. Assim, normalmente, precisamos nos basear em alguns pontos cruciais para o desenvolvimento do projecto, integrando, efectivamente, o usuário/cliente no processo de desenvolvimento de modo a acrescentar mais requisitos, aqueles que o sistema legado não satisfaz.

2.2.2.3. Mapear Elementos de Diagramas de Fluxos de Dados para Casos de Uso na Metodologia Escolhida

Um dos maiores problemas actuais no desenvolvimento de software está relacionado ao processo de transição entre metodologias. As organizações são obrigadas a mudar seus sistemas computacionais utilizando novas abordagens de desenvolvimento, as quais apresentam algumas vantagens tais como: maior flexibilidade, suporte automatizado para todo o processo de desenvolvimento, etapas e regras bem definidos, bem como possibilidade de uso efectivo do paradigma orientado a objectos. Contudo, na maioria das situações, existem informações importantes nos sistemas legados que devem ser mapeadas para novos sistemas gerados no uso de novas metodologias. Neste contexto, verifica-se, tipicamente, DFD desenvolvidos utilizando metodologias tradicionais precisam ser mapeados de forma mais efectiva e sistemática para a técnica de casos de uso das novas metodologias.

2.2.3. Ferramentas para Migração de Sistemas de Informação Legados

2.2.3.1. Ferramentas CASE

Uma ferramenta CASE é um aplicativo ou conjunto de especificações que auxiliam e automatizam as tarefas que compõem o processo de construção ou reengenharia de sistemas (Carlos, 2005). As ferramentas Case dividem-se em três categorias, a saber:

1. Lower Case – Ferramenta de Codificação (Front End);
2. Upper Case – Ferramenta de Análise, Projecto e Implementação;
3. Integrated Case – União de Upper Case e Lower Case.

Um dos componentes indispensáveis de uma ferramenta Case é a modelagem visual ou seja, a possibilidade de representar, através de modelos gráficos, o que está sendo definido. No nosso caso, análise orientada a objecto através da OOSE.

2.2.3.2. Família de Padrões de Reengenharia Orientada a Objectos – FaPRE/OO

A FaPRE/OO é uma família de padrões de reengenharia para gerar processos de engenharia orientados a objectos a partir de sistemas legados (Recchia et al, 2002). É composta por quatro fases, cada uma agrupando os padrões relacionados a situações similares da reengenharia. A figura 2.8 ilustra, graficamente, as fases e os padrões existentes em cada uma delas.

Fase 1: Modelar os dados do Legado: agrupa padrões que extraem informações a partir dos dados e do código fonte do sistema legado (se disponível), gerando o modelo de entidade e relacionamento (visão procedimental dos dados) e o modelo de análise do sistema Actual – diagrama de pseudo classes (visão orientada a objecto dos dados) (Recchia, 2002). Fazem parte desta fase os seguintes padrões: iniciar análise dos dados, definir chaves, identificar relacionamentos, e criar visão OO dos dados.

Fase 2: Modelar a Funcionalidade do Sistema: agrupa os padrões para obter a funcionalidade do sistema, criando modelos que recuperam as regras de negócio da organização contidas no sistema legado (Recchia, 2002). Esses padrões habilitam a equipa de reengenharia a obter um entendimento detalhado dos componentes (partes) do sistema, aprofundando, assim, sua compreensão sobre o sistema legado. Fazem parte desta fase os seguintes padrões: obter cenários, construir diagrama de use cases, elaborar descrição de use cases e tratar anomalias do sistema legado.

Fase 3: Modelar o Sistema Orientado a Objectos: agrupa padrões para obter o diagrama de classes e os diagramas de sequências do sistema através da interacção dos produtos obtidos pelos padrões das fases anteriores para obter o modelo de análise do sistema (Recchia et al, 2002). Fazem parte desta fase os seguintes padrões: definir as classes, definir os atributos, analisar hierarquias, definir métodos e construir diagramas de sequências.

Fase 4: Gerar o Sistema Orientado a Objectos: Agrupa padrões que completam o processo de reengenharia do sistema, transformando o sistema legado para o paradigma orientado a objectos. Fazem parte desta fase os seguintes padrões: definir a plataforma, converter o banco de dados, implementar os métodos, realizar melhorias na interface.

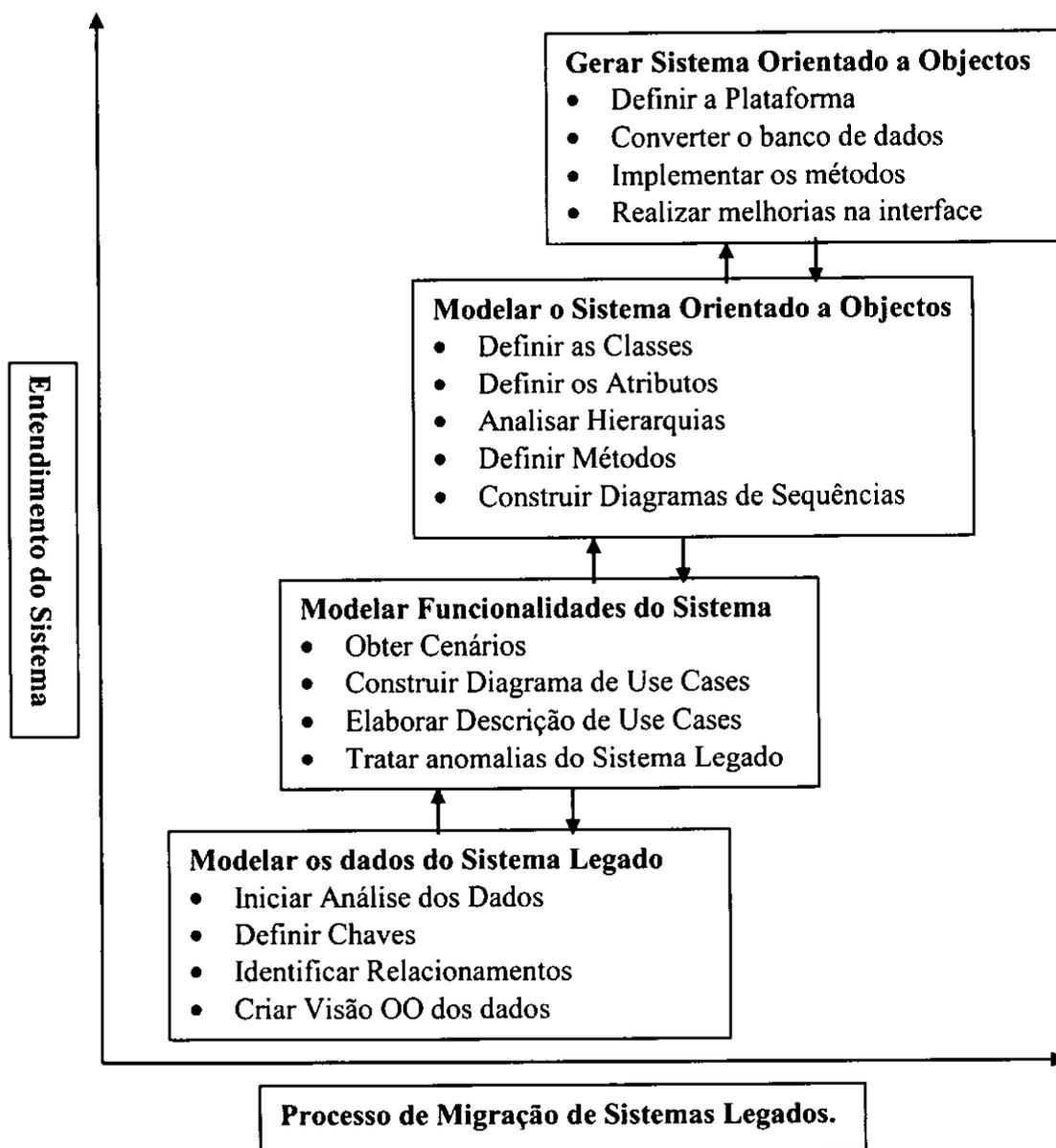


Figura 2.8 Processo de Migração de Sistemas de Informação Legados (Adaptado de Recchia et al)

Como se pode observar da Figura 2.8 alguns padrões terão como entrada a saída de algum padrão aplicado na fase anterior; na mesma figura pode-se, notar que o modelo é evolutivo, podendo-se de qualquer padrão, avançar ou retroceder após a sua aplicação.

2.2.3.3. Sistemas de Gestão de Base de Dados (SGBD)

Em um sistema legado, é mais provável que funcione num ambiente de sistema de ficheiro ou então num SGBD sem requisitos adequados da nova implementação. Neste caso, as estruturas de dados do

sistema fisicamente criadas de acordo com o diagrama de classes terão de ser implementadas em um SGBD relacional ou orientado a objectos.

Deve-se gerar uma base de dados, a partir dos arquivos que contêm os dados do sistema legado, usando ferramentas adequadas à linguagem de implementação e ao SGBD escolhidos para converter os arquivos de dados do sistema legado em tabelas de bases de dados relacional ou OO escolhido.

Na nossa implementação foi escolhido o SGBD relacional MySQL pelas razões apresentadas na secção 2.2.3.4. Neste caso, fez-se a conversão da base de dados Access 95 existente para o SGBD MySQL e foi usada a ferramenta de migração PhpMyAdmin.

2.2.3.4. O MySQL

MySQL é um SGBD relacional Open Source⁴ robusto, consistente, de alta confiabilidade e fácil de usar, que permite o armazenamento e recuperação de informação na base de dados de maneira mais eficiente e rápida (Welling e Thomson, 2001).

O MySQL é o mais popular SGBD, encontrando-se com mais de 10 milhões de instalações em varias organizações de todo mundo (MySQL LAB⁵,2007).

Para além de ser open Source e mais popular SGBD, o MySQL corre em mais de 20 plataformas (Sistemas Operativos), incluindo Linux, Windows, OS/X, HP-UX, AIX, Netware dando a mesma flexibilidade de manipulação de dados (MySQL LAB).

Segundo MySQL LAB (2007), estão abaixo descritos alguns dos pontos que fazem MySQL um SGBD popular e confiável:

- **Desenvolvimento de aplicações inclusivo** – o MySQL provê apoio inclusivo para todas as necessidades de desenvolvimento de aplicações. Dentro do MySQL estão embutidos procedimentos, funções, visões e cursores que podem ser usados no desenvolvimento de aplicações. Para além disso, MySQL provê motores de conexão ODBC (Open Database Connectivity) e JDBC

⁴ Open Source sua traduzido em Português, corresponde a Código Aberto, o que significa que é possível para qualquer individuo usar o software e modificá-lo à medida das suas necessidades. Mas existe uma licença comercial do MySQL se pretender desenvolver aplicação comercial que incorpore MySQL.

⁵ MySQL LAB é a empresa que desenvolve o SGBD MySQL.

(Java Database Connectivity) que permite a todo tipo de aplicação fazer uso do MySQL, portanto, não importa ser PHP, Java, .Net, C++, Perl, Python, existem API preparados para cada linguagem;

- **Segurança** – o MySQL possibilita a criação de sistemas portáteis, seguros, contendo privilégios e senhas (password) que permitem verificação baseado em usuário ou em máquina. O MySQL implementa fortes algoritmos de criptografia quando há transmissão de dados;
- **Volume de dados** – o MySQL foi construído para suportar grandes volumes de dados de tipos diferentes desde Integer, float, double, blob, date e mais, e possui um sistema de alocação de memória muito rápido baseado em processos;
- **Ferramentas de Migração** – MySQL dispõe de um driver ODBC para sistemas Windows, que nos permite conectar o Access com o próprio MySQL e passar informação facilmente. Para além disso existem ferramentas CASE preparadas pela MySQL LAB que permitem a migração dos diferentes SGBD incluindo Access para MySQL. Algumas dessas ferramentas são: PhpMyAdmin, MySQL Migration ToolKit.

2.2.3.5. Linguagens de Implementação

Depois da decisão da nova linguagem de implementação que atende os requisitos de orientação a objectos, segue a escrita dos métodos obtidos nas fase de análise e de desenho. Os critérios de escolha da linguagem de implementação são diversificados dependendo dos objectivos da nova implementação. Contudo, a equipa terá de ter domínio da linguagem para derivar os diagramas de sequencias em linhas de código.

A linguagem escolhida terá de permitir a concepção de sistemas que atendam, cada vez melhor, às necessidades dos usuários em relação, não apenas a critérios de funcionalidade (conjunto de tarefas desempenhadas pelo sistema) mas, também, à usabilidade e interfaces amigáveis.

Nesta implementação foi escolhida a ferramenta Visual Studio .Net, e em particular a sua linguagem de programação VB .NET. Esta escolha foi devido ao apresentado na secção 2.2.3.6. acrescida ao facto de o sistema legado em estudo ter sido construído com a linguagem Visual Basic for Applications (VBA) que é compatível a VB .NET.

2.2.3.6 Tecnologia Visual Studio .Net

É uma ferramenta de desenvolvimento de softwares que permite utilizar várias linguagens de programação, tais como: VB.NET, C#, C++ .Net, e J# (Capitão, 2002). Esta característica permite ao programador escolher a sua linguagem de programação preferida ou então escolher a linguagem que melhor se adequa ao programa que este pretende desenvolver. Para além da diversidade de linguagens, estão incorporadas desde as linguagens processuais até às orientadas a objectos e que permitem o desenvolvimento de aplicações tanto consola como GUI (Graphical User Interface) e até o desenvolvimento de aplicações WEB.

.Net Framework – Todas as linguagens que incorporam Visual Studio .Net partilham o mesmo módulo de programação e a mesma biblioteca de classes a .Net Framework.

.Net Framework é o conjunto de classes (biblioteca de funções e procedimentos) escritos pela Microsoft que se podem usar no desenvolvimento de aplicações (Prince e Gunderloy, 2002). .Net Framework substitui a API do Windows. Um outro aspecto importante que o .Net Framework trouxe é a uniformidade do compilador, isto é, todas as linguagens que compõem o Visual Studio .Net usam o mesmo compilador chamado CLR (Common Language Runtime), o que permite a partilha do código escrito em diferentes linguagens. Portanto, o programador poderá escrever o código na linguagem da sua preferência e, ao mesmo tempo, integrar o código de qualquer outra linguagem (Prince e Gunderloy, 2002).

Visual Basic .Net – é uma linguagem de programação totalmente orientada a objectos, criada pela Microsoft e distribuída com Visual Studio .Net.

Esta linguagem permite a construção desde aplicações de pequeno porte até as de grande porte. Possui mecanismos simples para a construção de interfaces de utilizador e é altamente compatível com VB6 e VBA.

CAPITULO III

METODOLOGIA DE PESQUISA

Capítulo III. Metodologia de Pesquisa

Neste capítulo, procede-se a descrição da metodologia de pesquisa que se optou para o alcance dos objectivos previamente definidos, incluindo a descrição das técnicas de pesquisa, colheita de dados e o estudo de caso.

3.1. Cenário e Estudo de Caso

3.1.1. SIL no SIS em Moçambique

Antes de 25 de Junho de 1975, existiam vários sistemas que funcionavam sem coordenação entre si, utilizando métodos diferentes de actuação, reflectindo uma divisão do sistema de saúde em componentes controladas por corpos administrativos, associações de classe, beneficências religiosas, empresas privadas (MISAU, 2003).

Entre 1975 e 1979, o SIS funcionou com uma orientação tipicamente centralizada e, só durante os primeiros anos da década oitenta, começou a primeira reestruturação do sistema marcada em 1983 com a chegada do primeiro micro-computador do sector da saúde (MISAU, 2003).

Em 1990, foi efectuada uma segunda revisão completa e, em Dezembro de 1991, o SIS foi introduzido à escala nacional na forma que ainda hoje mantém de base de informação estatística de saúde - enquadrada pelos diversos subsistemas que cobrem áreas específicas tais como Recursos Humanos, Manutenção, Gestão Financeira (MISAU, 2003).

Entre os anos 1992 e 1994, foi desenvolvido um programa automatizado denominado SISProg, representando o primeiro sistema nacional baseado em computador (Langa, 2005). Este sistema foi desenhado em duas versões, uma para o nível central e outra para o nível provincial. As duas versões estão baseadas no ambiente do sistema operativo MS DOS e escritos na linguagem Clipper (Langa, 2005). Outra característica deste sistema é que não suporta interface com mouse e pode correr em janelas MS DOS do Windows 95 mas não em Windows 98 com processador 80386 e 80486 (Nhampossa, 2006). O SISProg foi desenvolvido para automação de onze (11) programas de saúde tais como Vacinação, Planeamento Familiar, para reportar dados a partir dos distritos, via província até ao nível central.

Para além do SISProg, desenvolveu-se o SIM-ORGANIZER na plataforma Access 95, na província de Nampula, para automação de informação na área de manutenção provincial, que posteriormente veio a ser implementado sem sucessos na área de manutenção a nível nacional. A Tabela 3.1 ilustra os SIL existentes no sistema de saúde nacional.

Sistema	Plataforma	Tipo de Sistema	Função
SISProg	MS-DOS	Base de Dados dBaseIII	Controlo de 11 programas de Saúde a nível Provincial e Nacional
SIMP	MS Windows	Planilha Electrónica (Excel)	Planificação, Monitoria e Gestão integrada a nível Provincial e Nacional
SIP	MS Windows	Base de Dados Access	Gestão de Recursos humanos e formação
SIGETS	MS Windows	Base de Dados Access	Gestão de Stocks
SISTAFE	MS Windows	Base de Dados	Automação de administração financeira do Estado
SIM-ORG	MS Windows	Base de Dados Access 95	Automação de informação para a área de manutenção
FARMÁCIA	MS Windows	Base de Dados	Controlo de Medicamentos
BES	MS Windows	Base de Dados	Automatização do Boletim Epidemiológico Semanal
D03, D04	MS Windows	Planilha Electrónica (Excel)	Gestão de Infra-estruturas a nível Provincial
TUBERCULOSE	MS Windows	Planilha Electrónica (Excel)	Controlo de Tuberculose
HIV/SIDA	MS Windows	Base de Dados	Controlo de HIV/SIDA
CDS/ISIS	MS Windows	Base de Dados	Documentação

Tabela 3.1 Sistemas de Informação Legados desenvolvidos e em uso SIS (Adaptado: MISAU 2003 e Nhampossa 2006)

Como ilustra a Tabela 3.1 existem mais de dez (10) sistemas e aplicações que suportam o SIS. Contudo, estes sistemas não se comunicam entre si; a partilha de dados é praticamente inexistente de tal forma que cada sistema produz autonomamente dados e a informação, às vezes é duplicada, com critérios diferentes, sem padronização e, em consequência, dificilmente comparáveis, (problemas típicos de SIL).

3.1.2. Abordagens Anteriores aos SIL do MISAU

O SisProg antes referenciado foi desenvolvido por um técnico estrangeiro contratado pelo MISAU como uma tentativa de se obter uma solução mais integrada de sistema de informação de saúde (Nhampossa, 2006). Pode-se considerar como sendo uma das primeiras abordagens aos SIL a nível do MISAU, pois a ideia principal era integrar no SisProg dados provenientes de mais de dez sistemas ou programas. Porém, na prática, no SisProg, apenas dados de dois programas foram integrados, nomeadamente, Programa Nacional de Vacinação e Saúde Materno-Infantil. Como resultado, os restantes programas iniciaram seus projectos de software com o objectivo de tirarem vantagem do uso de tecnologias de informação e comunicação. Como consequência de tal procedimento, uma diversidade de sistemas paralelos ao SisProg foi desenvolvida e posta em uso (vide Tabela 3.1) o que contribuiu para reduzir a fiabilidade do SIS como tal e o incremento de sistemas legados.

Várias avaliações foram feitas com o objectivo de analisar o desempenho do SIS como um todo e produzir recomendações práticas para a reposição da fiabilidade do sistema. São exemplos a avaliação feita em 1999 pela Organização Mundial de Saúde (OMS), programa de desenvolvimentos do sistema de informação para saúde 2003-2005 (2010).

Entre as várias constatações das avaliações destaca-se: falta de uma definição precisa da política sobre o SIS, falta de cobertura satisfatória, dados atempados e retro-informação a todos os níveis, falta de troca de informação entre o SIS e os sub-sistemas de informação (exemplo SIM-ORGANIZER, recursos humanos).

É neste contexto que um Memorandum de Entendimento foi celebrado entre o MISAU, a UEM e a Universidade de Western Cape para o fortalecimento do SIS. Deste Memorandum resultou o programa *Helth Information System Program* (HISP). O HISP desenvolveu o sistema *Districtal Helth Information System* (DHIS) na perspectiva de contrariar os obstáculos impostos pela diversificação de sistemas que suportam o SIS. O DHIS é um sistema Open Source que agrega dados de todos os programas de saúde e que pode ser usado em todos os níveis da estrutura orgânica do MISAU, desde a unidade sanitária, distrito, província até a nível nacional. O DHIS, actualmente, não integra dados de manutenção.

A perspectiva deste trabalho é construir um sistema para manutenção e, posteriormente, integra-lo no DHIS para fazer com que o DHIS seja um produto completo para o sector nacional de saúde.

3.2. Método

Para o presente estudo, foi escolhido o tipo de pesquisa denominado “estudo do caso”. É definido por Yin (1989, P.13) como uma forma de fazer pesquisa social empírica ao investigar-se um fenómeno actual dentro de seu contexto de vida real, onde as fronteiras entre o fenómeno e o contexto não são claramente definidas e na situação em que múltiplas fontes de evidencia são usadas.

Os estudos de caso são, então, apropriados para pesquisas que possuem perguntas como “O quê?” “Como?” que exigem foco exploratório; em que não há controle do pesquisador sobre o comportamento dos eventos; e onde o foco são eventos contemporâneos, e não análise histórica apenas.

Realizou-se uma pesquisa qualitativa, exploratória – descritiva (sem carácter generalizador) para responder à questão da migração de SIL.

Conforme expõe Bogdan et all (1994), a pesquisa qualitativa tem como características:

1. O local natural como fonte directa de dados e o pesquisador como instrumento chave;
2. Carácter descritivo não quantificador;
3. Preocupação com o processo ao invés do produto;
4. Análise indutiva dos dados, ou seja, construção de abstrações como particularidades pesquisas em grupo ao invés de teste de hipóteses;
5. O significado como principal preocupação.

Segundo Neves (1996) apud Langa (2005), a pesquisa qualitativa ajuda no trabalho de construção do objecto estudado, facilita na descoberta de dimensões não conhecidas do problema e permite também formular e comprovar novas hipóteses.

A pesquisa qualitativa ajudou grandemente na aquisição de conhecimentos mais consistentes sobre o funcionamento do DM, bem como na formulação de ideias cada vez mais concisas sobre o problema que se propunha resolver.

São finalidades de uma pesquisa exploratória, sobretudo quando bibliográfica, proporcionar maiores informações sobre determinado assunto, facilitar a delimitação de um tema de trabalho, definir

objectivos ou formular hipóteses de uma pesquisa ou descobrir novo tipo de enfoque para o trabalho que se tem em mente (Andrade, 2001) apud (Langa, 2005).

A pesquisa exploratória auxiliou no entendimento das maneiras de utilização de informação para a tomada de decisões e avaliação de fontes de dados pelos gestores do DM/MISAU.

As fontes de dados usadas neste estudo foram essencialmente a revisão bibliográfica, entrevistas, observação directa e participativa e consulta documental, tais como: programa de desenvolvimentos do sistema de informação para saúde 2003-2005 (2010), relatório de avaliação do SIS volumes 01 e 02.

Na fase inicial do estudo foi promovido o fundamento teórico necessário para a sua realização. Foram feitos estudos (revisão de literatura) sobre engenharia de software, paradigma de orientação a objectos, metodologia de desenvolvimento de software, reengenharia de sistemas e SIL. O conteúdo desse estudo foi sucintamente descrito no capítulo 2.

Segundo D'Ascenção (2001), a entrevista é uma técnica de recolha de dados que consiste em uma conversação executada de forma planejada, sistemática e documentada. Geralmente é feita entre duas ou mais pessoas que têm interesses ou problemas em comum para obtenção de informação que estão na posse dos entrevistados fruto da sua experiência e convivência com a organização.

Para o presente trabalho, foram realizadas entrevistas estruturada e semi-estruturadas. A entrevista semi-estruturada foi realizada com o chefe do DM, pois o autor precisou de oferecer muita liberdade de esclarecimento dos assuntos pertinentes relativas a administração da manutenção (vide anexo B: Guia de Entrevista Semi-Estruturada).

As entrevistas estruturadas foram realizadas com os chefes das repartições de planificação e estatística, formação e normação e do gabinete técnico do DM para a obtenção do fluxo de actividades e o levantamento de processos da manutenção de equipamentos. (vide anexo A: Guia de Entrevista Estruturada).

Estas entrevistas permitiram a identificação dos problemas que envolvem a gestão da manutenção de equipamentos, assim como a colheita de diferentes sensibilidades e procedimentos em relação a gestão e controlo de equipamentos.

A observação directa consiste na verificação “in loco” de tudo o que acontece no ambiente de trabalho de forma sistemática e planejada, que permite uma verificação presente do ambiente e fluxo das informações (D’Ascensão, 2001).

Neste estudo, a observação directa/participativa foi realizada durante o período de Novembro de 2006 a Fevereiro de 2007. A escolha deste período foi devido ao trabalho intenso que se regista nas US por ser um período festivo e de grande movimentação da população e, conseqüentemente, o trabalho de manutenção é intenso segundo um dos elementos por mim entrevistado.

Embora tivesse previamente marcado a realização da observação directa/participativa no DM e nas unidades de manutenção dos hospitais centrais do país, esta só foi possível ser realizada no DM e no sector de transportes do MISAU. Nas unidades de manutenção não foi possível devido às limitações de ordem financeira e de ordem temporal do autor.

Nos locais onde foi possível a observação directa/participativa, o autor fez parte da equipa de trabalho durante o horário laboral, onde este obteve melhor sensibilidade acerca da rotina de manutenção de equipamentos e inteirou-se da administração da manutenção de equipamentos do MISAU. Para além disso, permitiu, também, a verificação pessoal e o respectivo registo dos factores que influenciam ou que podem influenciar no sistema em estudo como: interdependência e inter-relação com os demais subsistemas do MISAU.

O processo de construção do modelo de software seguiu os princípios básicos dos métodos de desenvolvimento OO descritos no capítulo 2 (secção 2.1), com auxílio das ferramentas de transição de sistemas legados descritos no capítulo 2 (secção 2.2). Adoptou-se a metodologia de desenvolvimento OOSE, que propõe a definição de vários modelos para a visualização dos vários aspectos do software. O desenvolvimento do software foi realizado em um processo incremental e iterativo, utilizando os resultados de um ciclo de desenvolvimento como subsídio para a realização do próximo ciclo (o capítulo 4 descreve este processo).

Após a finalização da arquitectura do modelo através do refinamento dos diagramas que compõem as especificações em OOSE, efectuou-se o desenho da base de dados para a implementação em quaisquer sistemas de gestão de base de dados relacional. Particularmente para a esta implementação foi escolhido o SG BD MySQL (o capítulo 4 trata desta implementação).

A consulta da documentação disponível no DM, para além de permitir ampliação do conhecimento do sistema em estudo em relação ao histórico dos problemas identificados e alternativas adoptadas para solucioná-los, auxiliou, fundamentalmente, no desenho de interfaces com utilizador e permitiu mapear a realidade visual com a implementação real. Para o desenho de interface e codificação foi usado o Microsoft Visual Studio .NET através da sua linguagem de programação VB .NET.

Para o teste do modelo de sistema, elaborou-se um conjunto de testes separadamente a cada modulo que compõe o sistema durante a construção e um teste global do modelo foi feito após a construção envolvendo a equipa do DM (vide anexo D). Estes testes determinaram que o modelo é útil na administração das actividades do DM.

CAPITULO IV

MODELAÇÃO DO SISTEMA DE INFORMAÇÃO PARA GESTÃO DE EQUIPAMENTOS MÉDICOS

Capítulo IV: Modelação do Sistema de Informação para Gestão de Equipamentos Médicos

Neste capítulo apresenta-se a descrição, análise e desenho do sistema em estudo usando grande parte dos conceitos da metodologia OOSE.

4.1. Descrição do sistema

O MISAU é o órgão principal do Sistema Nacional de Saúde (SNS) cuja missão é de promover e preservar a saúde da população moçambicana, promover e prestar cuidados de saúde de boa qualidade e sustentáveis, tornando-os, gradualmente, acessíveis a todos os moçambicanos com equidade e eficiência (MISAU, 2003). Para o cumprimento com zelo desta missão, o MISAU é auxiliado por vários subsistemas que suportam o Sistema de Informação para Saúde (SIS). Um dos subsistemas que auxilia o funcionamento do SIS é o Sistema de Informação para Gestão de Equipamentos Médicos (SGEM) que funciona no DM.

O DM possui uma estrutura orgânica própria, a nível das províncias, existem as unidades de manutenção subordinadas ao DM, chamadas Secção Provincial de Manutenção (SPM). Para além destas, os hospitais centrais possuem, também, unidades de manutenção, chamadas Secção de Manutenção Hospitalar (SMH). O DM participa em todo o ciclo de vida dos equipamentos desde a aquisição, manutenção até ao abate.

Quando se pretende adquirir um equipamento médico específico, a Comissão Técnica de Equipamento e Consumíveis (COTEC), composta por colectivo de médicos, representantes do DM e colectivo a nível central do MISAU reúne-se para a definição das especificações técnicas do equipamento a ser adquirido e, de seguida, é lançado o concurso (caso seja necessário) nacional ou internacional para aquisição do equipamento.

Depois de aquisição, o departamento de logística e aprovisionamento do MISAU codifica o equipamento de modo que ele seja identificado unicamente a nível nacional, posteriormente o equipamento é enviado a US que irá utilizar. A direcção da US comunica a unidade de manutenção da sua província a existência do novo (ou não inventariado) equipamento de modo que a unidade de

manutenção também registe para planear futura manutenção do mesmo e procede a instalação do equipamento se necessário.

No caso de avaria de um equipamento, a US comunica à UM a ocorrência da avaria. Por sua vez, a UM abre uma Folha de Obra⁶ (FO) e faz análise técnica da avaria. Desta análise, três situações podem ocorrer:

1. O equipamento avariado tem um contrato válido de manutenção com o seu fornecedor;
2. A avaria registada no equipamento supera a capacidade técnica da UM;
3. A UM possui capacidades técnicas para solucionar o problema.

No primeiro caso, a UM, em coordenação com o fornecedor, faz análise do custo de trabalho e depois de aprovação deste custo o fornecedor inicia as actividades de manutenção.

No segundo caso, a UM comunica o DM, o qual poderá enviar os seus técnicos a nível central ou pode optar pelo lançamento de concurso às entidades externas prestadoras de serviços de género e procede-se com os trabalhos de manutenção.

No último caso, a UM procede com os trabalhos podendo fazer requisição de peças específicas ao DM em casos de necessidade.

Nos três casos, depois de conclusão das actividades, a US faz a aprovação do trabalho o que marca o fim da actividade de manutenção.

Em cada trimestre as unidades de manutenção devem enviar um relatório das actividades realizadas ao DM e este por sua vez, faz a compilação para um relatório nacional e envia-o para MISAU e outras entidades que se relacionam com o DM.

A Figura 4.1 mostra as entidades envolvidas e suas relações no sistema em estudo.

⁶ Ficha de registo das actividades de instalação e manutenção de equipamentos

Requisitos do Sistemas

- Inventariar as US e os equipamentos;
- Excluir ou Editar US e equipamentos;
- Registrar as actividades de manutenção;
- Ter, a todo o momento, um conhecimento detalhado do estado e da localização das US e dos equipamentos do MISAU;
- Consultar o histórico dos equipamentos;
- Monitorizar e controlar os custos de manutenção;
- Gerar relatórios periódicos das actividades de manutenção;

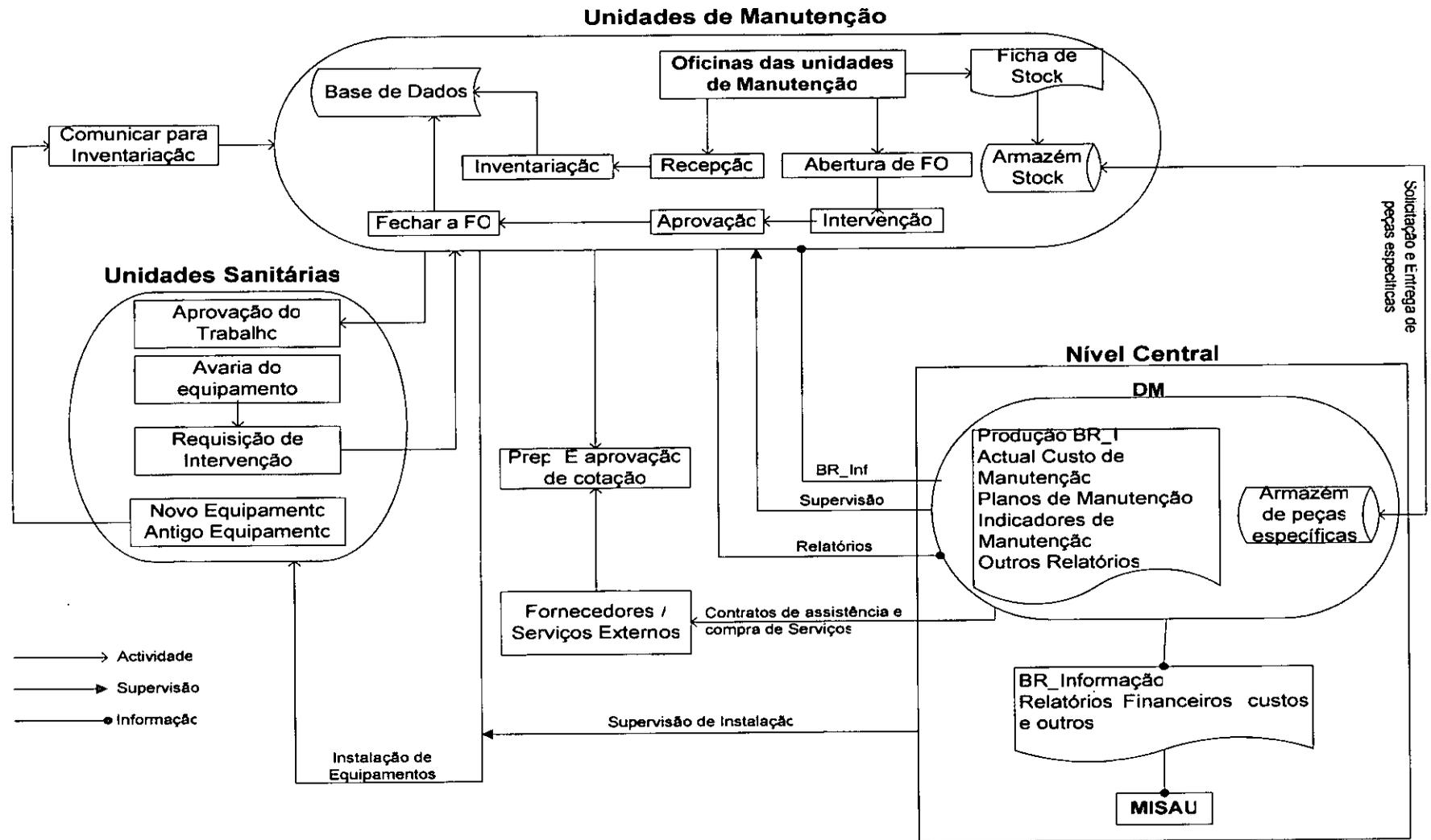


Figura 4.1 Descriçãc do Sistema

4.2. Modelo de Análise

4.2.1. Nomenclatura de Equipamentos Médicos.

A padronização da nomenclatura e da codificação é um trabalho de carácter básico e permanente para um sistema de informação; para os equipamentos médicos; existe um padrão de nomenclatura definido pela Emergency Care Research Institute (ECRI) chamado Universal Medical Device Nomenclature System (UMDNS) para permitir maior facilidade de comunicação e troca de informação entre os administradores de equipamentos mesmo que se expressem em idiomas diferentes (Pallikarakis et al, 1996).

A UMDNS propõe que cada grupo genérico distinto de equipamentos ou produtos médicos seja identificado por cinco dígitos hifenizados (por exemplo, o código 11-129 está sempre associado aos Defibrillator / Monitor). O número hifenizado de cinco dígitos não contém mais informações para além de definir o tipo de equipamento, ficando ao critério de cada órgão acrescentar informação a este código dependendo do seu contexto.

Para a nomenclatura de equipamentos, a UMDNS propõe que seja feita através de uma construção hierárquica de três palavras separadas por vírgulas em que:

- A primeira palavra identifica o equipamento de maneira geral (exemplo: Aspirador);
- A segunda e a terceira palavras definem a função e o tipo de equipamento, respectivamente (exemplo, Cirúrgico, Vertical). Neste exemplo, a nomenclatura do equipamento seria: **Aspirador, Cirúrgico, Vertical.**

4.2.1.1 Nomenclatura das US

Para se adequar às instituições internacionais, o DM tem de padronizar a nomenclatura dos bens para facilitar a troca de informações, quer a nível nacional quer a nível internacional, usando, deste modo a nomenclatura proposta por UMDNS.

Cada sistema sanitário tem sua estrutura de referência que é composta de órgãos de controlo (ministério e direcções periféricas), órgãos executivos (US⁷ com o seu nível de atendimento) e órgãos formativos e de apoio. Em Moçambique, a estrutura com a sua própria nomenclatura e nível de atendimento é a seguinte: (Tabela 4.1)

Posto de Saúde	Hospital Especializado
Centros de Saúde	
Hospital Rural / Distrital	Centros de Formação/CRDS
Hospital Geral / Provincial	DDS
Hospital Central	DPS/MISAU

Tabela 4.1 Lista de níveis de atendimento e administrativos no SNS. (Fonte: MISAU, 2003)

4.2.1.2. Nomenclatura de Equipamento Geral e Específico

Para o equipamento geral e específico, incluindo os meios circulantes, o DM terá de seguir devidamente a UMDNS, mas para casos em que se torna difícil a identificação das três componentes da nomenclatura, o DM deve adoptar uma estrutura de nomenclatura alternativa, para tal, terá de usar os termos gerais tais como:

Aparelho: para designar todo o equipamento biomédico que entra em contacto directo ou indirecto com o doente separado por vírgula e seguido pelo nome do equipamento. (ex.: Aparelho, Electroforese);

Máquina: para todo o equipamento de apoio hospitalar (ex.: Máquina, Lavar Roupa).

4.2.2. Codificação de Equipamentos Médicos.

Tal como a estandardização da nomenclatura facilita a comunicação entre os Técnicos da Manutenção do DM e os fabricantes dos aparelhos ou seus agentes autorizados à escala nacional e internacional, também, uma codificação bem definida e generalizada deve permitir o seguinte:

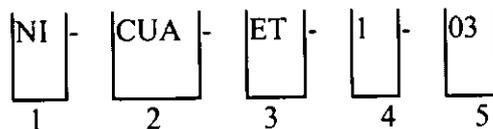
- Localizar univocamente um item, seja a US como um equipamento dentro do país;
- Localizar univocamente um equipamento dentro de uma US;
- Classificar univocamente o tipo do item.

⁷ São consideradas Unidades Sanitárias, para efeitos de manutenção, todas as infra-estruturas pertencentes ao MISAU, isto é, inclui-se as DPS, DDS, Centros de Formação e outras.

Portanto, é importante definir um código unívoco para poder identificar todos os aspectos acima referidos.

4.2.2.1. Codificação de US

Um código definido com a estrutura a seguir mostrada permite identificar uma US dentro dum país, um item dentro da US.



O código acima contém as seguintes informações:

1. Dois caracteres alfanuméricos para representar a divisão administrativa primária (províncias);
2. Três caracteres alfanuméricos para identificar a divisão administrativa secundária (distritos);
3. Dois caracteres alfanuméricos para representar a divisão administrativa terciária (posto administrativo);
4. Um campo numérico (de 0 a 9) para indicar o nível da estrutura dentro da organização do sistema sanitário;
5. Dois caracteres numéricos para representar o número de ordem da estrutura sanitária do mesmo nível dentro uma dada divisão administrativa terciária.

4.2.2.2. Codificação de Equipamentos

Para a codificação de equipamento, o DM interrelaciona-se directamente com a Direcção Nacional do Património (DNP), pois todos os equipamentos a nível do MISAU possuem um único identificador nacional (nº do património) de modo que se consiga obter a localização dos mesmos. Nesta codificação, os primeiros cinco números seguem a codificação da UMDNS que identificam o grupo ou o tipo de equipamento, seguidos do código da US a que pertence o equipamento e, posteriormente, uma sequência de cinco dígitos que indicam o número ordenado do equipamento dentro da US.

Por exemplo: para um " Aspirador, Cirúrgico, Vertical " do Centro de Saúde de Etatara na Província de Niassa poderá ter o código seguinte:

11 - 129 - NI - CUA - ET - 1 - 03 - 00001

Onde:

- 11-129 – tipo de equipamento, neste caso aspirador;
- NI – província de Niassa;
- CUA – distrito de Cuamba;
- ET – posto administrativo de Etatara;
- 1 – centro de saúde;
- 03 – ordem do centro de saúde no posto administrativo;
- 00001 – número de ordem do equipamento dentro da US.

4.2.3. Construção do Modelo de Requisitos.

Nesta fase, a preocupação é com o funcionamento do sistema de modo a satisfazer os requisitos anteriormente identificados, para tal tem-se como base os pressupostos da metodologia OOSE.

4.2.3.1. Identificação de actores

- **US** – a nível do DM as US correspondem aos hospitais, unidades administrativas e serviços educacionais sob tutela do MISAU. São entidades que lidam directamente com o equipamento e a sua missão é de usar correctamente os equipamentos e comunicar as avarias.
- **Funcionário** – é um individuo do MISAU afecto ao DM que faz o uso e administração de dados dos equipamentos.
- **Administrador** – faz a gestão de dados de manutenção de equipamento, funcionários, fornecedores, incluindo os custos.
- **Fornecedor** – a designação alternativa deste actor é serviços externos; recebe pedidos tanto de fornecimento de equipamentos como de fornecimento de serviços de manutenção dos mesmos.

4.2.3.2. Identificação de use cases para cada actor

US

- **Comunicar a existência de novo equipamento** – quando a US recebe ou adquire um novo equipamento deve comunicar a UM para que registre no sistema para planear futuras manutenções;
- **Requisitar uma Intervenção** – quando um equipamento avaria, a US requisita uma intervenção a UM para a manutenção do mesmo;
- **Consultar estado do equipamento** – depois da US enviar o equipamento para a manutenção, consulta o estado de manutenção do mesmo;
- **Aprovar o trabalho de manutenção** – após o fim do trabalho de manutenção, a unidade deve aprovar o trabalho feito pela equipa de manutenção.

Funcionário

- **Inventariar US** – registo dos dados das US para que se possa conhecer a localização geográfica e inclui a modificação de dados da US;
- **Inventariar equipamento** – registo de dados dos equipamentos, suas especificações técnicas;
- **Abrir F.O.** – registo do início das actividades de manutenção de um equipamento, seja ele inventariado ou não desde que necessite de uma manutenção preventiva ou correctiva;
- **Fechar F.O.** – registo do fim de actividades de manutenção.

Administrador

- **Manter informação sobre equipamento** – monitorização de dados de equipamentos para garantir a disponibilidade dos mesmo quando solicitados;
- **Manter informação sobre fornecedores** – registar dados dos fornecedores de equipamentos e outros serviços de manutenção;
- **Manter informação sobre funcionário** – registo de dados dos funcionários de manutenção, incluindo modificação e anulação dos mesmos;
- **Manter informação sobre custos de manutenção** – registo dos custos referentes á manutenção dos equipamentos.

Fornecedor

- **Receber pedido de fornecimento de equipamento e serviços** – receber novos pedidos de fornecimento de equipamento e/ou serviços;
- **Fornecer produtos e serviços** – fornecer produtos e serviços solicitados.

Administrador e Funcionário

- **Consultar dados de equipamento** – consulta dos diferentes dados dos equipamentos;
- **Consultar dados dos funcionários** – consulta dos diferentes dados dos funcionários;
- **Emitir Relatórios** – produzir diferentes tipos de relatórios necessários para administração e controlo de equipamentos;
- **Emitir Indicadores de Manutenção** – emissão de indicadores de avaliação de desempenho institucional e de disponibilidade de equipamentos;
- **Emitir Histórico de Equipamento** – o histórico do equipamento pode ser, por exemplo, o número de vezes em que o mesmo avariou, o custo de manutenção do mesmo.

Funcionário e Fornecedor

- **Reparar equipamento** – se estiver celebrado um contrato de fornecimento e assistência técnica de alguns equipamentos, os fornecedores fazem a reparação dos equipamentos e são supervisionados pelo DM.

A Figura 4.2 ilustra o diagrama de *use case* do sistema em estudo, mostrando, para cada actor os seus respectivos use cases .

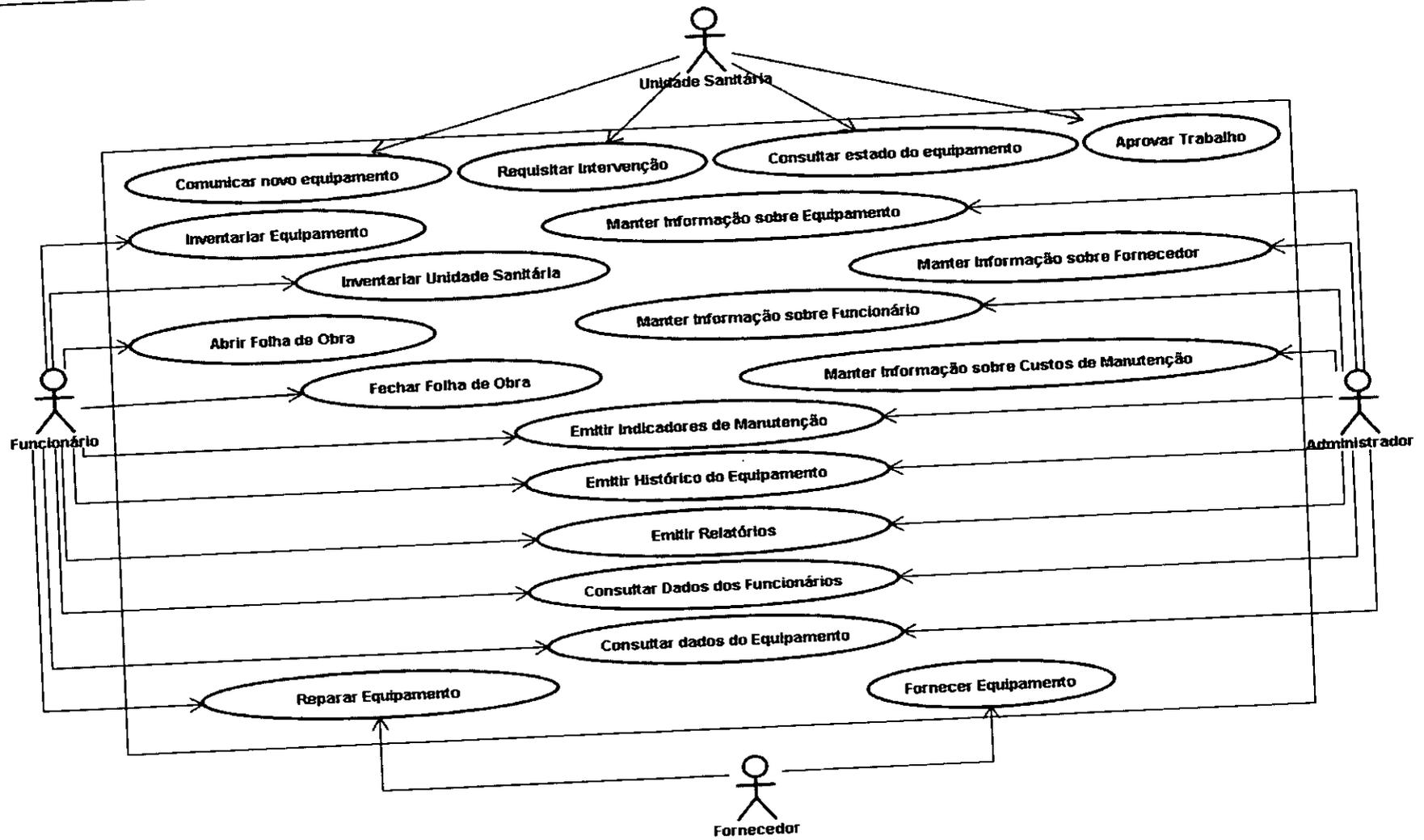


Figura 4.2 Diagrama de Use Cases do sistema em estudo

4.2.3.3. Descrição de Use Cases

A seguir apresentam-se as descrições detalhadas de alguns use cases, mostrando como e quando cada actor envolvido para cada *use case* interage com o sistema.

Comunicar a existência de um novo⁸ equipamento

Um equipamento inventariado, ou seja, registado pelas unidades de manutenção, facilita o controlo e monitorização quando ele está avariado, por isso, quando um equipamento não for inventariado, a US deve comunicar a unidade de manutenção através do envio do código do equipamento atribuído pelo departamento de logística e aprovisionamento do MISAU e outros dados de caracterização do equipamento.

Requisitar uma Intervenção

Depois de avaria de um equipamento, a US preenche uma ficha de pedido de intervenção (reparação da avaria) contendo, entre outros dados, o número do inventário, a data de ocorrência da avaria e a descrição técnica da avaria. Esta ficha é enviada a UM. O referido pedido também pode ser efectuado por telefone. Em ambos os casos o sistema regista este pedido alocando-o um número de pedido que é enviado à US que procedeu a requisição dos serviços de manutenção.

O sistema codifica e regista o tipo de avaria que ocorre (caso seja um novo tipo de avaria), para manter o histórico de avarias que frequentemente ocorrem em equipamentos e como se resolvem as similares.

Este use case serve para o controlo das entradas dos equipamentos às UM, das obras efectuadas e pendentes, e, também, permite identificar os tipos de avarias que ocorrem em equipamentos.

Consultar estado do equipamento

Depois de a US ter enviado um equipamento para manutenção, recebe um número de pedido, (use case requisitar uma intervenção) e, com base neste número, o sistema verifica se já foi aberto uma FO para o pedido. Em caso afirmativo, o sistema determina o real estado do equipamento que pode ser, por exemplo, reparação interrompida por falta de peça sobressalente. Em caso negativo, o sistema responde que ainda não iniciaram os trabalhos de manutenção para a satisfação do pedido.

⁸ O termo novo equipamento refere-se ao equipamento ainda não inventariado pelas unidades de manutenção.

Inventariar Equipamento

Quando se adquire um novo equipamento, este deve ser registado como um património do MISAU.

Este registo, primeiramente, é feito na DNP e é alocado um identificador único nacional, o qual fica disponível para o uso com outras entidades do MISAU, como o DM:

1. O sistema pede o número de identificação do equipamento;
2. O funcionário fornece o número;
3. O sistema verifica se existe algum equipamento com o número introduzido;
4. Se não existir algum equipamento com o número fornecido, o sistema pede os dados adicionais e as especificações técnicas do equipamento;
5. O funcionário fornece os dados e o sistema regista o equipamento;
6. Se existir algum equipamento com o número introduzido, o sistema reporta esta duplicação e envia-se à DNP para as devidas correcções e o sistema aloca um identificador temporário válido só no DM e regista o equipamento após o fornecimento das especificações técnicas do mesmo.

Inventariar US

Como se referiu anteriormente as US para o DM correspondem aos hospitais, unidades administrativas e serviços educacionais sob tutela do MISAU.

O fluxo neste processo é semelhante ao inventariar equipamento.

Folha de Obra ou Ficha de Obra (FO)

É uma ficha que contém as informações dinâmicas sobre o(s) equipamento(s); nesta ficha são descritas informações como: o tipo de intervenção técnica efectuada, quem a executou, o tempo gasto na reparação do(s) equipamento(s), os materiais empregues e o custo total da intervenção.

Abrir FO

O sistema verifica as requisições efectuadas aquelas que estão pendentes, isto é, aquelas requisições que ainda não foram satisfeitas, portanto ainda não houve a abertura da FO.

O funcionário identifica aquela requisição para qual pretende abrir a FO, podendo, neste momento, ocorrer dois cenários:

1. a requisição é sobre um equipamento inventariado no DM;
2. a requisição é sobre um equipamento não inventariado (ou infra-estrutura).

No primeiro caso, o sistema regista no histórico do equipamento, entre outros dados: a data da ocorrência da avaria, o técnico que irá realizar o trabalho e, é depois mudado o estado do equipamento para avariado.

No segundo caso, o sistema abre a FO e emite um alerta para a necessidade de registo do equipamento como um património do MISAU.

Em ambos os casos considera-se uma folha aberta no momento em a requisição fica alocada a um técnico para a realização dos trabalhos e quando se regista a data de início das actividades de manutenção.

Fechar FO

É um processo que ocorre no final de uma actividade de manutenção na seguinte ordem:

1. O sistema verifica as folhas de obra abertas e ainda não fechadas;
2. O funcionário selecciona a FO que pretende fechar;
3. O sistema regista o relatório de manutenção que inclui entre outros dados, o custo total de manutenção, as intervenções técnicas realizadas, eventualmente informações de como proceder durante a manutenção do equipamento.

Considera-se uma FO devidamente fechada, quando se preenche a data de entrega do equipamento à US requisitante e o sistema imprime três exemplares da FO devidamente preenchida. Estes três exemplares são enviadas ao sector requisitante para a confirmação (através da assinatura do responsável do sector requisitante). A distribuição destes três exemplares assinados é feita da seguinte forma: um exemplar fica no sector requisitante, outro fica nos serviços administrativos da US e o último fica na unidade de manutenção.

O sistema veda as possíveis actualizações de uma FO fechada, pois, pela regra do DM, uma FO fechada não pode, de modo algum, ser alterado os seus dados.

Emitir Relatórios

Um relatório deve ser capaz de fornecer informações, de maneira clara e sucinta, para permitir a avaliação da situação de um sector, o trabalho desenvolvido e o fornecimento de elementos suficientes e necessário para a intervenção correcta em eventuais situações alarmantes. Portanto, o sistema poderá emitir diferentes tipos de relatórios dependente do nível e do órgão destinatário. Desta maneira, na Figura 4.3, a seguir apresentada, estão reportados os diferentes tipos de relatórios e respectivos destinatários.

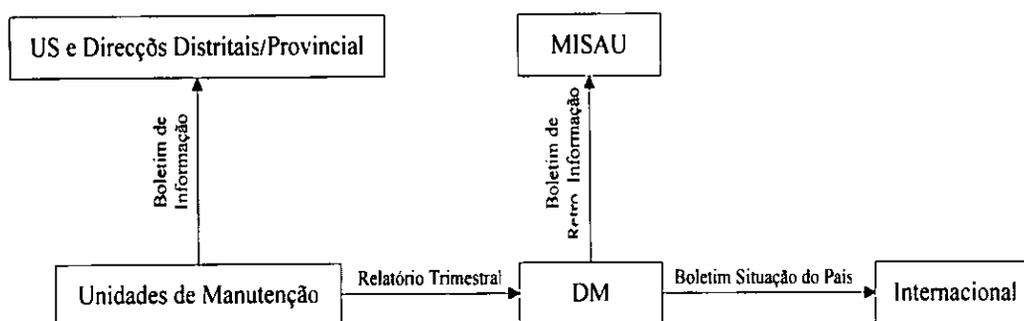


Figura 4.3 Relatórios a serem emitidos pelo sistema e seus destinos⁹

- **Relatório Trimestral:** São os relatórios técnicos que servem para a análise sobre a gestão de equipamentos no país. Este relatório é emitido pelas unidades de manutenção e é destinado ao DM;
- **Boletins de Informação (B_Info):** a nível das SPM e SMH são produzidos estes boletins que reportam todos os aspectos relacionados com as actividades de manutenção. Estes boletins deverão ser elaborados clara e concisamente de modo a que os responsáveis da Direcção Provincial de Saúde (DPS) e da administração hospitalar possam compreendê-los e interpretá-los.
- **Boletim de Retro-Informação (BR_I):** Boletim produzido pelo DM que fará uma síntese e uma comparação entre os B_Info's periféricos e introduzirá, eventualmente, outras informações produzidas a nível central, tais como certas indicações e procedimentos relacionados com a Nomenclatura e Codificação.

⁹ O sistema fornece grande parte de informação a este relatório, a outra informação vira de outras fontes para um relatório completo.

Para além dos relatórios acima descritos, o sistema permitirá produzir vários outros relatórios de acordo com a informação requerida, sendo de destacar:

- Informações referentes à FO;
- Informações sobre a nomenclatura, marcas e modelos;
- Informações sobre o financiamento;
- Informações sobre o pessoal;
- Demonstrações gráficas.

Emitir Indicadores de Manutenção

Os indicadores são a forma de avaliação institucional mais usada no sector de saúde. Para a avaliação do sector de manutenção foram seleccionados seis grupos de indicadores. Embora a maioria destes indicadores sejam relacionados aos equipamentos, alguns deles são calculados para veículos, edifícios e instalações técnicas especiais e para o desenvolvimento da própria instituição. Os grupos dos indicadores são:

- Indicadores de desenvolvimento institucional;
- Indicadores do estado do património;
- Indicadores do nível de equipamento disponível;
- Indicadores de actividade;
- Indicadores económico-financeiros;
- Indicadores de qualidade.

Para emitir cada um destes indicadores é necessário uma recolha e análise cuidadosa dos dados referentes ao inventario e das actividades de manutenção. Alguns indicadores são abaixo descritos:

Percentagem de Equipamentos em Funcionamento

Indica a percentagem dos equipamentos em funcionamento. Tal percentagem é calculada em relação aos equipamentos inventariados obtido pela divisão dos equipamentos em funcionamento pelo total inventariado num dado período e numa US.

Custo de Intervenção / Custo de Reposição

É a relação entre o custo da reparação e o preço de reposição. O custo de manutenção do período em referência para cada equipamento é comparado ao valor de reposição (compra de um novo) do mesmo. Este indicador serve de suporte para a tomada de decisão de abate ou substituição do equipamento ou viatura.

Tempo de resposta

Indica quanto tempo, em média, dura uma intervenção técnica. O calculo exacto deste indicador deve ser a diferença entre a data do pedido de intervenção e a data da fecho da FO.

A Tabela 4.2 mostra os vários indicadores de manutenção incluindo a sua representação simbólica e nível de decisão de interesse.

Uso das Metodologias Orientadas a Objectos para Migração de Sistemas de Informação Legados:
Estudo de Caso Sistema de Informação para Gestão de Equipamentos Médico

Nº	Grupo de Indicadores	Tipo de Indicador	Símbolo	Nível Interesse
01	Desenvolvimento Institucional	Índice de grau de desenvolvimento institucional	DESV_INST	Cimeiro
02	Estado do Património	Percentagem de Equipamento em funcionamento	%EF	Cimeiro
		Percentagem de bens inventariados / bens existentes	%BI/BE	Cimeiro
03	Nível Tecnológico	Valor de grupo tecnológico per cama per nível de atendimento	Niv_TEC	Cimeiro
04	Actividades	Unidade Standard de Manutenção	USM	Operacional
05	Económico-Financeiros	Custo de Serviço / Custo de Reposição	CS/CR	Cimeiro
		Execução Orçamental	%Exec	Cimeiro
		Dotação de Manutenção / Dotação de Funcionamento	DM/DF	Cimeiro
		Gastos de Manutenção / Valor activo a Manter	GM/AM	Cimeiro
		Gastos dos Serviços Externos / Gastos do Sistema	GSE/GS	Cimeiro
		Dotação de Manutenção / Valor activo a manter	DM/AM	Cimeiro
		Valor dos Stocks / Valor dos bens a Manter	VS/BM	Cimeiro / Oper.
06	Qualidade	Actividade Realizada/Capacidade Instalada	Y1	Cimeiro / Oper.
		Pedidos de Intervenção Satisfeitos/Recebidos	Y2	Cimeiro / Oper.
		Tempo Parado (Downtime)	Y3	Cimeiro / Oper.
		Reparações Repetidas (Repeat Repair)	Y4	Cimeiro / Oper.
		Tempo de Resposta	Y5	Cimeiro / Oper.
		Execução Orçamental	Y6	Cimeiro / Oper.
		Produtividade Laboral	Y7	Cimeiro / Oper.
		Marcador de Qualidade	Y8	Cimeiro / Oper.
		Custo de Intervenção / Custo de Reposição	Y9	Cimeiro / Oper.
		Actividade Realizada / Capacidade Necessária	Y10	Cimeiro / Oper.

Tabela 4.2 Resumo de indicadores do Sistema de Manutenção. (Fonte: MISAU, 2003)

4.2.3.4. Especificação de Interface

A parte final do modelo de requisitos em OOSE consiste na especificação de interface do sistema. No sistema em estudo os elementos fundamentais de interface são: vídeo (monitor), para a visualização de dados e teclado e mouse, para entrada de dados. A Figura 4.4 apresenta o “lay-out” da tela principal do sistema.

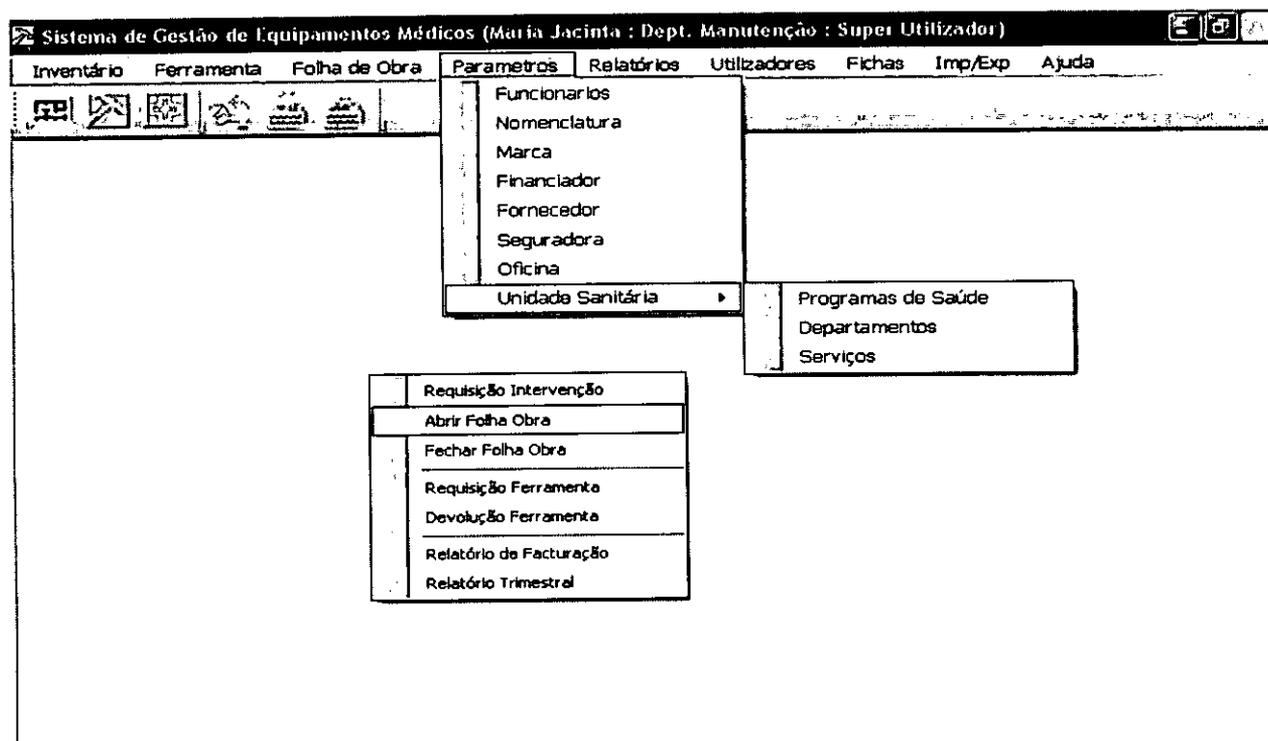


Figura 4.4 Lay-out de Tela Principal do Sistema

4.2.4. Construção do modelo de análise

Nesta fase, a metodologia OOSE propõe a identificação e descrição de classes que compõem o sistema em desenvolvimento. Da análise feita, identificou-se dez (10) classes fundamentais que a seguir se apresenta a descrição do propósito de cada uma delas; as associações de cada uma destas classes estão apresentadas no modelo de objecto apresentado na figura 4.5. De referir que as classes auxiliares não estão aqui apresentadas e, também, as classes interface e coordenadora não aparecem no modelo de objecto apresentado na Figura 4.5, pois elas fazem parte nos modelos de objecto que descrevem cada use case.

Classe de Interface:

- Interface de Usuário – cada actor comunica-se com o sistema através de uma interface, e cada interface representa uma classe que executa métodos responsáveis pela comunicação entre os usuários e sistema.

Classe de Controle:

- Coordenador – definida como a classe que coordena os acessos da base de dados.

A metodologia OOSE obriga a construção de um modelo de objectos e a sua descrição textual para cada use case do sistema. Para exemplificar, vamos considerar os use cases inventariação de equipamentos que desde já chamaremos de *inventariação* e abrir e fechar FO que passaremos a chamar de *folha de obra*.

4.2.4.1. Use Case Inventariação

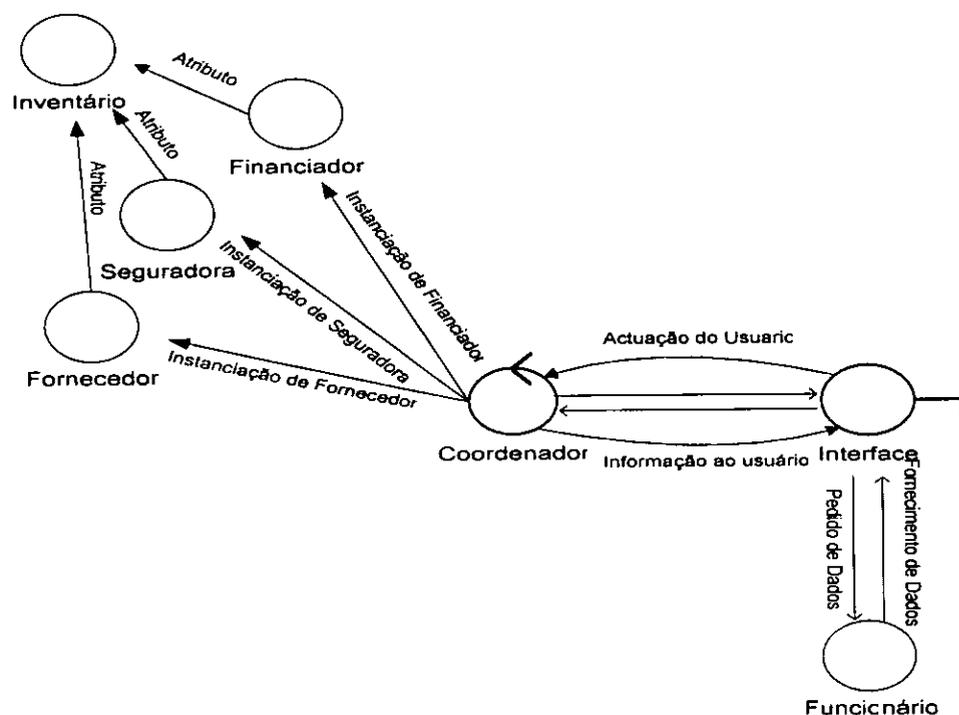


Figura 4.6 Modelo de objectos referente a use case "Inventariação"

Descrição de Use Case Inventariação (Figura 4.6)

1. A instância *Interface* cria uma instância *Coordenador* através da solicitação do método de criação de instâncias da classe *Coordenador*;
2. O *Coordenador* solicita à *Interface* o código do *Inventário*;
3. A *Interface* solicita este código ao *funcionário*;
4. O *Funcionário* fornece este código;
5. A *Interface* transmite este código ao *Coordenador*;
6. O *Coordenador* verifica a disponibilidade deste código;
7. Se o código não estiver disponível, isto é, estiver sendo usado com outro equipamento, o *Coordenador* envia uma mensagem à *Interface* que referencia a duplicação do código de identificação e esta reenvia esta mensagem ao *Funcionário*;
8. Se o código estiver disponível, o *Coordenador* solicita os dados adicionais do equipamento a ser inventariado à *Interface*, e, depois de receber os dados o *Coordenador*, invoca os métodos de instanciação das classes *Seguradora*, *Financiador* e *Fornecedor*, estas instanciações fazem parte dos atributos da classe *Inventário*;
9. Por último, o *Coordenador* invoca o método de instanciação da classe *Inventário* e cria um objecto encapsulado do tipo *Inventário* e chama o seu método de registo de inventário que é responsável pela persistência de dados na base de dados.

4.2.4.2. Use Case Folha de Obra

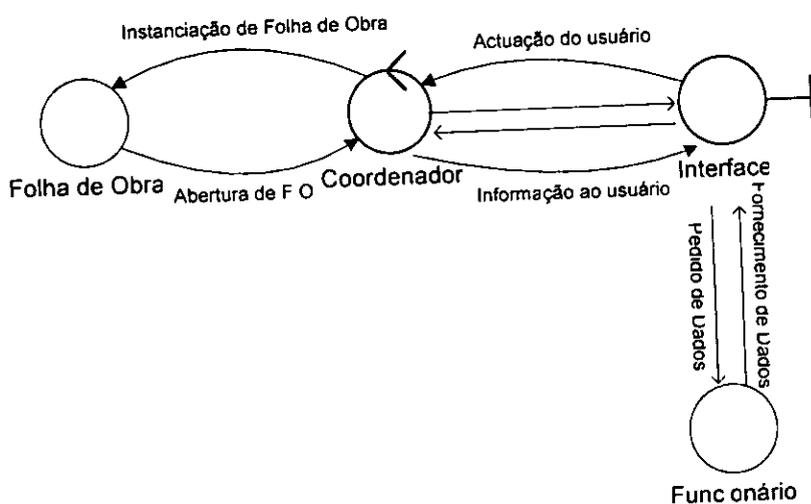


Figura 4.7 Modelo de objecto referente a Use Case "Folha de Obra"

Descrição de use case F O (Figura 4.7)

1. O *Funcionário* solicita a partir da *Interface* a abertura da FO;
2. A instância *Interface* cria uma instância *Coordenador* através da solicitação do método de criação de instâncias da classe *Coordenador*;
3. O *Coordenador*, através do seu método de pesquisa, procura todas as solicitações de intervenção registadas e envia a lista ao *Interface*;
4. Por sua vez, a interface solicita ao *Funcionário* a selecção do pedido pelo qual pretende abrir a FO;
5. Após a selecção do pedido pelo *Funcionário* a *Interface* envia esta selecção encapsulando dados adicionais ao *Coordenador*;
6. O *Coordenador* cria uma instância da FO e invoca o seu método para o registo de FO e regista a abertura da FO.

4.3. Projecto do Sistema

4.3.1. Construção do Modelo de Projecto

Esta fase, de OOSE, consiste na construção de um modelo de objecto semelhante ao construído no modelo de análise, com uma diferença de estrutura simbólica em que as classes passam a ser designadas por blocos e representados por rectângulos (vide Figura 4.8) e, depois, segue a construção do diagrama de interacção. Pode-se optar por representar o diagrama de blocos em apenas um diagrama ou mesmo representar um diagrama para cada use case, na nossa implementação construiremos um único diagrama bloco de generalização do sistema.

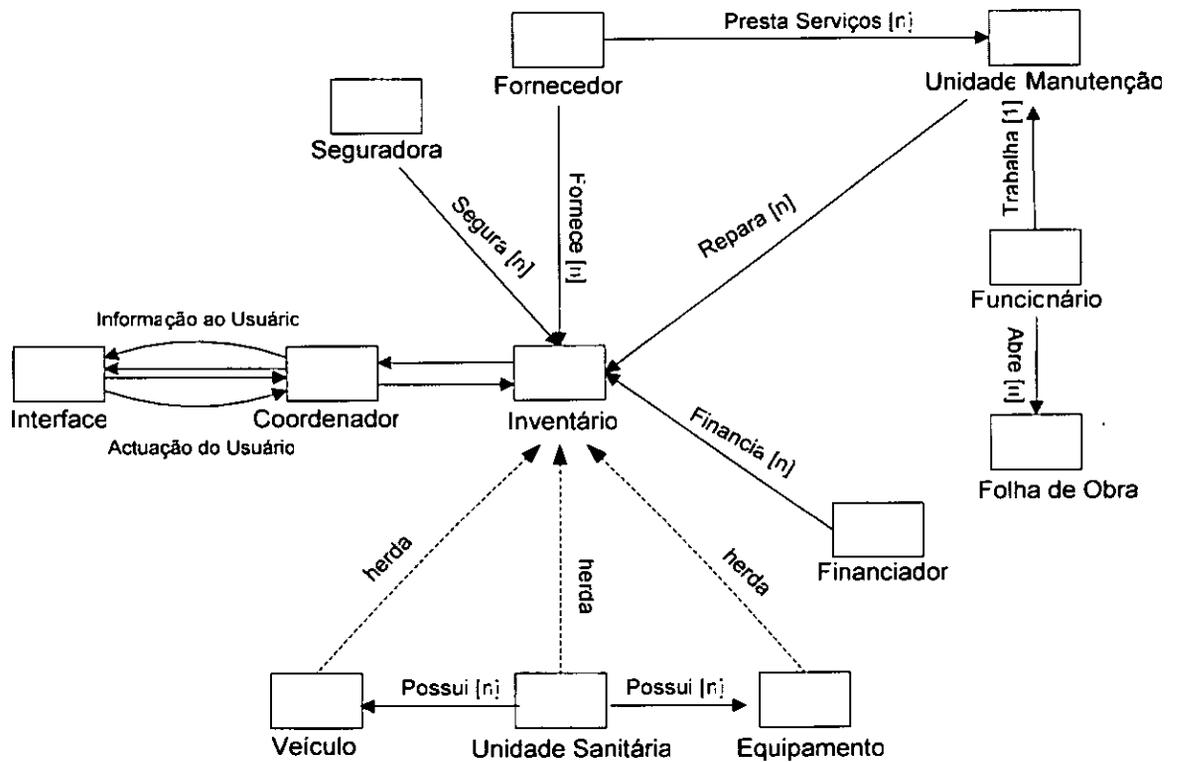


Figura 4.8 Modelo de blocos do sistema

A Figura 4.9 e a Figura 4.10 descrevem os diagramas de interacção para Use Case Inventariação e FO, respectivamente.

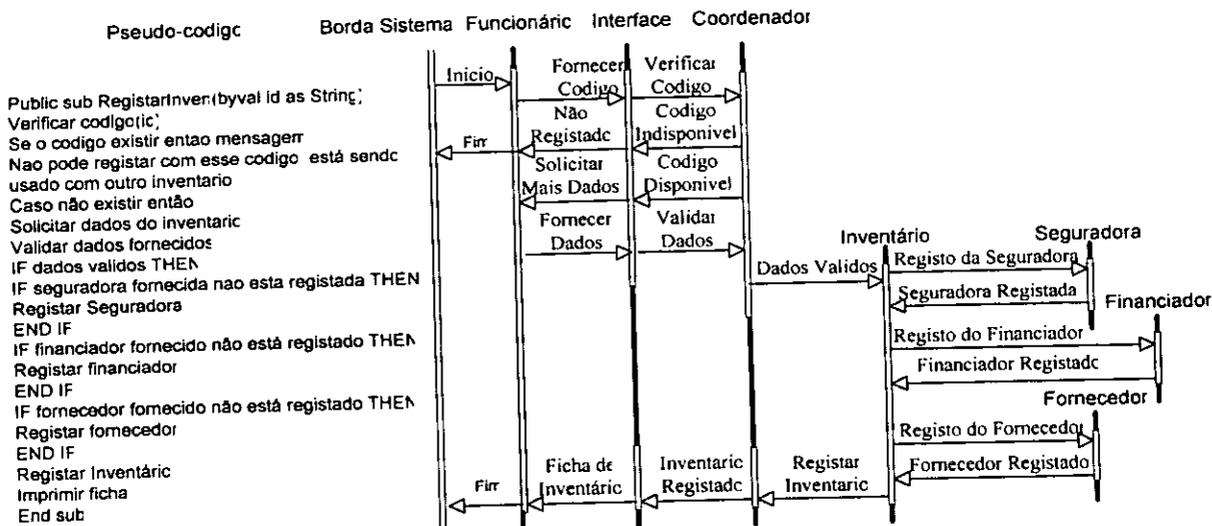


Figura 4.9 Diagrama de interacção para use case "Inventariação"

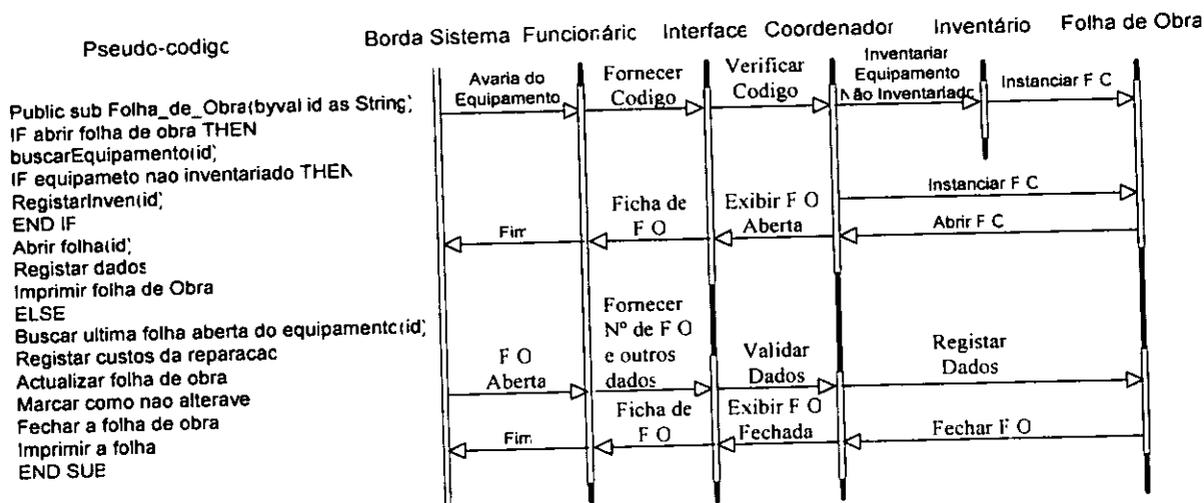


Figura 4.10 Diagrama de interacção para use case "Folha de Obra"

4.3.2 Modelo de Implementação

Esta fase, de OOSE, consiste no desenho da base de dados e na codificação das classes que compõem o sistema na linguagem de programação escolhida, neste caso em VB.NET.

A Figura 4.11 apresenta o modelo físico da base de dados do problema em estudo, o qual foi construído a partir das classes identificadas durante as fases de análise e de desenho e foram refinadas para uma realidade de implementação no SGBD MySQL.

Neste modelo, os rectângulos representam as tabelas físicas do modelo relacional da base de dados, em que a parte sombreada contém o nome da tabela; os atributos, a negrito e sublinhados representam as chaves primárias das tabelas e as setas os relacionamentos das tabelas.

Uso das Metodologias Orientadas a Objectos para Migração de Sistemas de Informação Legados:
Estudo de Caso Sistema de Informação para Gestão de Equipamentos Médico

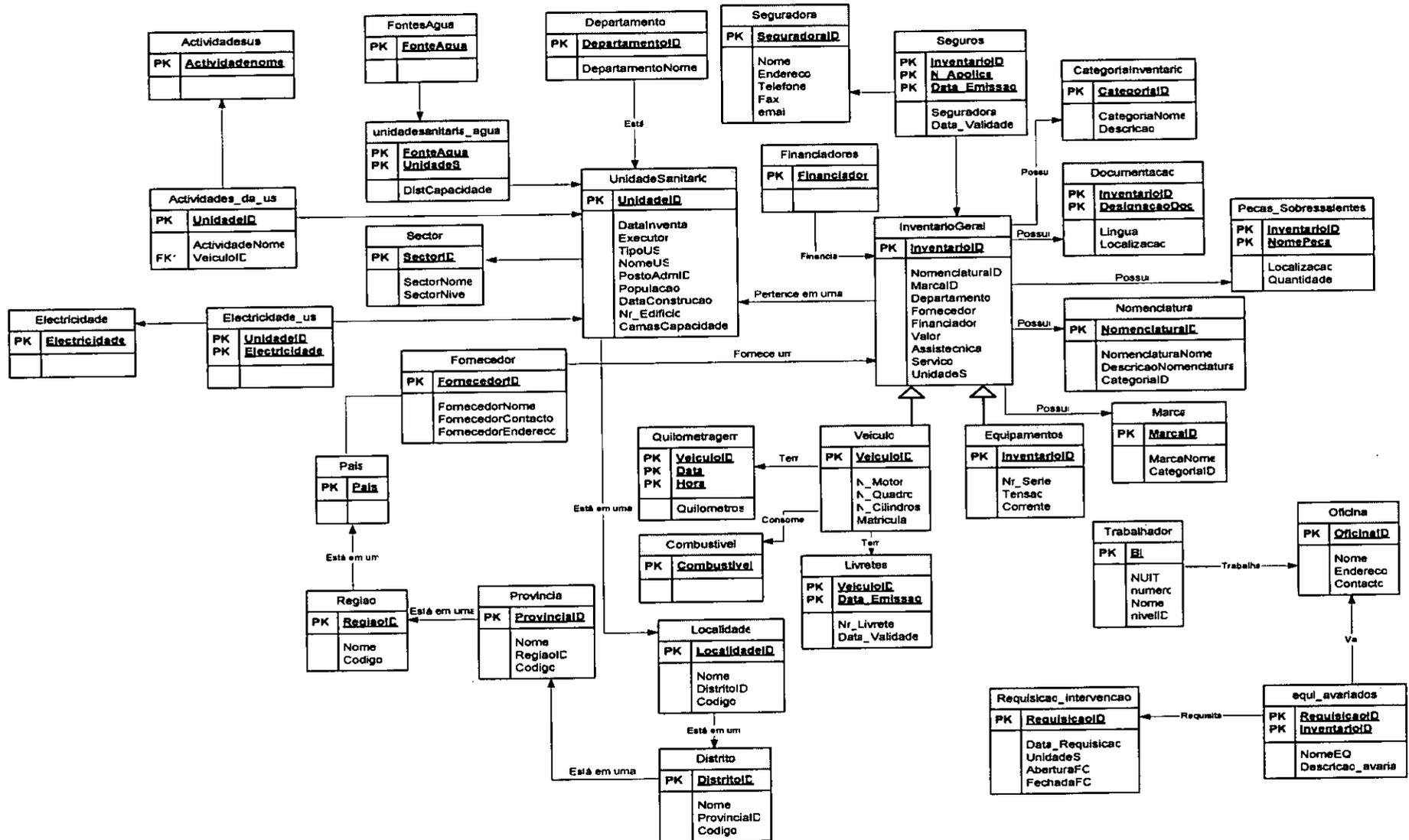


Figura 4.11 Modelo físico de base de dados do sistema em estudo

4.3.2.1. Arquitectura de implementação.

Contrariamente ao sistema legado, onde a aplicação e a base de dados funcionam interligadas na máquina onde estiver instalada a aplicação, sistema de computador pessoal (Stand-alone), a implementação aqui proposta segue uma arquitectura diferente. Apesar da simplicidade da arquitectura de computador pessoal, não é ideal para implementação no DM. Segundo o nosso entrevistado, é preciso que cada técnico de manutenção tenha acesso ao sistema em qualquer gabinete do departamento. Neste contexto, a arquitectura cliente-servidor foi implementada neste sistema em três (3) camadas: a camada de interface com utilizador, a camada de comunicação e negócio e a camada de dados (vide a Figura 4.12). Esta implementação das três camadas possibilita o isolamento da aplicação com a fonte de dados o que permite maior estabilidade do sistema.

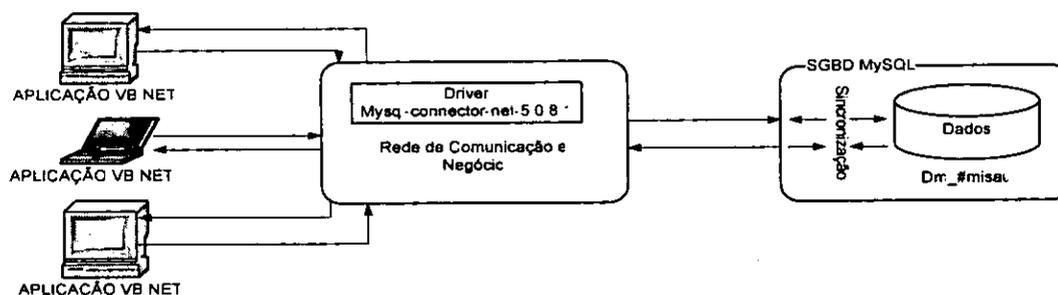


Figura 4.12 Arquitectura de Implementação do Sistema em Estudo

Como se pode observar na Figura 4.12, para a implementação do sistema num ambiente de produção é preciso uma estrutura de rede local, um servidor de Base de dados e as máquinas cliente com sistema operativo Windows XP SP2, Windows 2000 SP4 ou Windows Server e .Net Framework 2.0¹⁰ ou maior.

A outra alternativa de implementação, menos aconselhada pelo autor num ambiente de produção devidos às especificações de desenho, é ter a aplicação e a base de dados na mesma máquina (Arquitectura centralizada).

4.3.2.2. Modelo de Teste

A fase de análise e de desenho desta modelação foi validada e aceite pelo DM. O teste de sistema foi feito módulo a módulo durante o desenvolvimento e um teste global foi feito pela equipa do DM. O foco do teste foi fundamentalmente sobre, a interface, a estrutura (apresentação e organização de

¹⁰ A aplicação já vem com .Net Framework 2.0 embutido para actualização nas maquinas cliente que não tenham ou tenham versão de .Net Framework inferior a 2.0

dados), a funcionalidade (satisfação dos requisitos) e a comparação com o sistema legado (SIM - ORGANIZER) (vide o anexo E modelo de teste). Estes testes determinaram que o sistema responde às necessidades do DM, é de fácil uso e é melhor quando comparado com o SIM - ORGANIZER.

4.3.2.3. Informação Geral Sobre os Custos de Implementação

Do ponto de vista de *hardware*, para implementação do sistema proposto os custos seriam relativamente baixos, pois o DM já possui uma rede de dados local funcional com máquinas clientes, faltando apenas um servidor de base de dados.

Do ponto de vista de *software* é necessário sublinhar que tanto para o desenvolvimento da base de dados (*back end*) como da interface (*front end*) existem ferramentas grátis disponíveis; é o caso do SGBD *MySQL* para a construção da base de dados e da versão Express Edition do Visual Studio 2005 para o desenho da interface.

O sistema proposto no presente trabalho foi desenvolvido com base nas ferramentas grátis referidas. É necessário porém sublinhar que para o caso específico do desenvolvimento da interface do sistema dificuldades de vária ordem foram encontradas, impostas pelas limitações técnicas da versão não licenciada

4.3.2.4. Benefício Organizacional após Implementação do Sistema

O grau de conscientização dos benefícios de um sistema de informação varia muito entre os trabalhadores da organização. Mas acreditamos que a faixa deste grau inicia no patamar onde os trabalhadores têm total desconhecimento sobre a matéria até onde o trabalhador tenha sofrido uma prejuízo irreparável no seu negócio. Este, com certeza, por “ter sentido na pele” os problemas inerentes à falta de organização e documentação dos seus processos, passa a valorizar adequadamente os seus benefícios.

Ao documentar os seus processos, a organização tem a oportunidade de capturar e tornar transparentes as boas práticas usadas pelos seus trabalhadores. É sob esta forma que a organização realmente toma posse destas boas práticas, incorpora-as no seu património e torna-as acessíveis aos outros trabalhadores.

Para além das vantagens gerais anteriormente referenciadas, ao implementar este sistema, o DM ganhará uma vantagem competitiva a nível dos diferentes órgãos do MISAU, pois, permitirá o fornecimento de informação em tempo útil aos órgãos decisórios do MISAU. Contribuindo, deste modo, para a satisfação dos seus trabalhadores e o melhoramento da prestação dos cuidados sanitários à comunidade.

CAPITULO V

CONCLUSÕES E RECOMENDAÇÕES

Capítulo V: Conclusões e Recomendações

Este capítulo apresenta as conclusões a que o autor chegou a respeito do tema em estudo e da questão – problema de pesquisa.

O desenvolvimento de sistemas requer a utilização de métodos e ferramentas que possibilitem avaliar se o sistema em causa está sendo construído em conformidade com os padrões convencionais. O presente trabalho propôs-se a estudar a aplicabilidade de metodologias orientadas a objectos para a migração de sistemas de saúde legados, seguindo as práticas convencionais descritas na literatura e usadas na indústria de desenvolvimento de software.

Assim as conclusões deste trabalho são descritas em três áreas temáticas conforme se segue:

Metodologias de desenvolvimento de sistemas

- As metodologias de desenvolvimento de sistemas facilitam a construção de equipas de arquitectura de sistemas, maior comunicação e aproveitamento dos modelos de engenharia construídos na base destas metodologias.
- Estas metodologias providenciam uma linguagem apropriada para a comunicação dos membros de equipas de desenvolvimento de sistemas e uma apresentação padrão dos elementos do sistema em desenvolvimento. Com efeito, os modelos desenvolvidos com base nestas metodologias de análise e de desenho de sistemas, tornam-se de fácil leitura, interpretação e manutenção dada a padronização que lhes caracteriza e a cultura de documentação que elas impõem.
- Neste trabalho, foram descritas diferentes metodologias de análise e de desenho orientados a objectos; destacou-se a metodologia OOSE que serviu como base de análise e de desenho do sistema aqui apresentado. A escolha da metodologia OOSE foi satisfatória (ver secção 4.3.2.2) na modelação do sistema, pois, permitiu representar todas as características desejadas da aproximação de solução do problema.
- Apesar de a metodologia OOSE ser muito referenciada por vários autores na área de desenvolvimento OO, a falta de literatura específica sobre esta metodologia foi notória e constituiu constrangimento para o seu aprofundamento.

Sistemas Legados

- Os sistemas legados servem como base de sustentação para nova especificação; estes não devem ser ignorados quando se pretende desenvolver um novo sistema.
- As metodologias de desenvolvimento de sistemas, ferramentas case e a família de padrões de engenharia orientada a objectos são ferramentas indispensáveis para reestruturação dos sistemas legados para uma abordagem OO.

Sistema de Gestão de Equipamentos Médicos

- O modelo proposto foi construído usando padrões de engenharia de sistemas em todas as fases de desenvolvimento (análise, desenho e implementação, etc.), conforme especificado em OOSE, de modo a responder todos os requisitos e as regras de negócio relativas à manutenção de equipamentos do MISAU.
- A separação entre a base de dados e a interface do utilizador permite uma melhor performance e facilidade de uso do sistema.
- A implementação deste sistema tem potencial para servir como suporte integral a uma gestão e controlo dos equipamentos e actividades de manutenção a nível das unidades de manutenção do MISAU. O DM, nestes termos dispõe de uma ferramenta que lhe permitirá fornecer informação útil e atempada aos decisores do MISAU a todos níveis, ganhando uma vantagem competitiva em relação aos outros órgãos e, contribuindo deste modo para a melhoria na prestação dos cuidados sanitários.

Recomendações

- O sistema aqui proposto tem aceitação a nível do DM; Porém, recomenda-se que sua implementação integral seja precedida de:
 1. Uma avaliação pelo sector de sistema de informação do MISAU ou outra entidade com competência comprovada;
 2. Construção de uma equipa dedicada que integre entre outros um administrador de sistema. Este terá o papel de garantir a manutenção do sistema e o funcionamento do ponto de vista técnico num esforço coordenado com os diferentes níveis de utilização;
 3. Aquisição da versão licenciada e profissional do Visual Studio 2005(8) que é uma ferramenta mais completa com recursos apropriados para uma implementação mais segura e estável.
- Recomenda-se que estudos posteriores sejam conduzidos para o desenvolvimento e disponibilização da versão WEB do sistema. Como tal e para melhor aproveitamento do código do sistema e integração com VB.NET, recomenda-se ASP.NET como tecnologia a usar.

CAPITULO VI

BIBLIOGRAFIA

Capítulo VI: Bibliografia

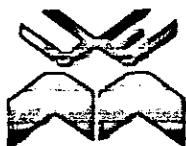
- (MISAU, 2003) MISAU (2003), Programa de Desenvolvimento do Sistema de Informação para Saúde 2003-2005 (2010), http://www.misau.gov.mz/pt/misau/dpc_direccao_de_planificacao_e_cooperacao/informacao_para_a_saude/documentos_chave_do_sistema_de_informacao_para_a_saude (Data de acesso 23/02/07)
- (D'ascensão,2001) D'ASCENÇÃO, M. L. C. (2001). Organização, Sistemas e Métodos, 1ª Edição, Atlas, São Paulo.
- (Langa, 2005) LANGA, L.B. (2005), Uso das Tecnologias de Informação e Comunicação no Sector da Saúde em Moçambique: SIS.D e sua Implementação nas Províncias de Gaza e Inhambane, Tese de Licenciatura, UEM, Maputo
- (Nhampossa, 2006) NHAMPOSSA, J. L. (2006), Strategies to Deal With Legacy Information Systems: A Case Study From The Mozambican Health Sector, Universidade de Oslo, Noruega
- (HAWRYSZKIEWYCZ, 1998) HAWRYSZKIEWYCZ, I. (1998), Systems Analysis and Design, 4ª Edição, Printice Hall, Sydney.
- (Bennet et al, 2002) BENNET, S., McROBB, S., FARMER, R. (2002), Object – Oriented Systems Analysis and Design Using UML, 2ª Edição, McGraw-Hill, London
- (Silva e Santander, 2000) SILVA, P. S. e SANTANDER, V. F. A (2000), Uma Proposta de Evolução em Sistemas Legados, Universidade Estadual do Oeste do Panamá.
- (Leite e Júnior, 2007) LEITE e JÚNIOR (2007), Uma Introdução ao Software Baseado em Objectos. Monografia disponível em <http://br.monografias.com/trabalhos/uma/uma.shtml#historico> (20/07/07)

- (Papajorgji e Pardalos, 2006) PAPAJOGRJI, P. J., PARDALOS, P. M. (2006), Software Engineering Techniques Applied to Agricultural Systems and Object-Oriented and UML Approach, Springer
- (Dori, 2002) DORI, D. (2002), Object Process Methodology a Holistic Systems Paradigm, Springer
- (Fumo, 2001) FUMO, T. G. S. (2001), A MOO e a UML aplicadas na construção de um sistema de apoio à gestão de uma unidade sanitária, Tese de Licenciatura, UEM, Maputo
- (Henderson, 2000) HENDERSON, P. (2000), Systems Engineering For Business Process Change, Springer
- (Rodriguez e Ferrante, 2001) RODRIGUEZ, M. e FERRANTE, A. J. R. (2001), Tecnologia Informação e Gestão Empresarial, e-Papers
- (Carlos, 2005) CARLOS, J. (2005), Ferramentas Case, [http://www.imasters.uol.com.br/artigo/3048/uml/ferramentas case/](http://www.imasters.uol.com.br/artigo/3048/uml/ferramentas_case/) (Data de Acesso 08/01/2008)
- (Recchia et al, 2002) RECCHIA, E. L., LEMOS, G. S., PENTEADO, R., BRAGA, R. T. V. (2002) Padrões para o Processo de Engenharia Avante na Reengenharia Orientada a Objectos de Sistemas Legados Procedimentais. <http://www.cin.ufpe.br/~sugarloafplop/articles/ww/PadroesEngenhariaAvante.pdf> (Data de Acesso 08/01/2008)
- (Recchia , 2002) RECCHIA, E. L., (2002) Engenharia Reversa e Reengenharia Baseada em Padrões, Dissertação de Mestrado, Universidade Federal de São – Carlos, São Paulo.
- (Welling e Thomson, 2001) WILLING, L. e THOMSON L. (2001), PHP and MySQL Web Development, SAMS, USA

- (MySQL LAB, 2007) MySQL LAB (2007) Site Oficial da MySQL (<http://www.mysql.com>) data de acesso (23-05-2007)
- (Capitão, 2002) CAPITÃO, V. (2002), Programação em Visual Basic .Net, FCA-Editora de Informática, Lisboa.
- (Prince e Gunderloy, 2002) PRINCE, J. e GUNDERLOY, M. (2002), Visual C# .Net, SyBex, USA.
- (O'Neill e Nunes, 2001) O'NEILL, H., NUNES, M. (2001), Fundamental de UML, FCA – Editora de Informática, Lisboa
- (Pallikarakis et all, 1996) PALLIKARAKIS, N., ANSELMANN, N., PERNICE, A. (1996), Information Exchange for Medical Devices, IOS Press.
- (Yin, 1989) YIN, R. K. (1989), Case Study Research – Design and Methods, Sage Publications, London.
- (Bogdan et all, 1994) BOGDAN, R., BIKLEN, S. K., ALVAREZ, M. J., VASCO, A. B, SANTOS, S. B., BAPTISTA, T. V. M, SANTOS, S. B, (1994) Investigação Qualitativa em Educação: Uma Introdução à Teoria e aos Métodos, Porto Editora, Portugal.
- (Yourdon, 1991) YOURDON, E., COAD, P. (1991) Object – Oriented Design, Prentice Hall.
- (Jacobson et all, 1992) JACOBSON, I., CHRISTERSON, M., JONSSON, P., e OVERGAARD G. (1992), Object – Oriented Software Engineering – A Use Case Driven Approach, ACM Press
- (Ricardo, 1996) RICARDO, P. S., (1996), Avaliação de Metodologias de Análise e Projeto Orientados a Objectos Voltadas ao Desenvolvimento de Aplicações, sob Ótica de sua Utilização no Desenvolvimento de Frameworks Orientados a Objectos, Tese de Pós-Graduação, Universidade Federal do Rio Grande do Sul, Porto Alegre.

ANEXOS

ANEXO A: Guião de Entrevista Estruturada



UNIVERSIDADE EDUARDO MONDLANE
FACULDADE DE CIENCIAS
DEPARTAMENTO DE MATEMÁTICA E INFORMÁTICA

GUIAO DE ENTREVISTA

Este guião de entrevista visa recolher informação relativa ao funcionamento a estrutura orgânica e administração de equipamentos médicos do Departamento de Manutenção do Ministério de Saúde, no âmbito do trabalho de Licenciatura em Informática do estudante **Eurico José**, do Departamento de Matemática e Informática da Universidade Eduardo Mondlane, cujo o tema é:

Uso das Metodologias Orientadas a Objectos para Migração de Sistemas de Informação

Legados:

Estudo de Caso: Sistema de Gestão de Equipamentos Médicos.

A informação que for colhida será tratada com confidencia e somente para o fim a cima descrito.

Local: Departamento de Manutenção

Hora de início: ____:____

Data ____/____/____

Hora de fim : ____:____

Nome do Entrevistado: _____

Cargo / Função : _____

Repartição / Sector: _____

QUESTÕES PARA ENTREVISTAS

I. Instituição

1. O que é DM?
2. Qual é a missão do DM?
3. Como funciona o DM?
4. Como o DM se relaciona com outros órgãos do MISAU? E com outras entidades fora do MISAU?
5. Como avalia o nível de Implementação de TI no DM?

II. Equipamento

1. Qual o nível de crescimento de aquisição de equipamentos hospitalares no sector de saúde?
2. Quando é que o DM entra no processo de controlo de equipamento?
3. A nomenclatura de equipamentos hospitalares no DM segue alguma regra? Se sim, qual?
4. Como é que o DM Identifica os equipamentos?
5. Que informação é relevante para um equipamento?
6. Quais os procedimentos de obtenção de informação relativo ao equipamentos?
7. Como essa informação é tratada ?
8. Quais os mecanismos de disseminação dessa informação? Para quem? Qual a periodicidade?
9. Qual a utilidade dessa informação?

III. Intervenções

1. Quem pode solicitar o DM para uma Intervenção?
2. Como é que o DM é solicitado?
3. O DM faz intervenção para todo tipo de equipamento hospitalar?
4. Qual é a quantidade média de intervenções que o DM faz durante um mês?
5. Como é mantido o registo das intervenções actualmente?
6. Se não é registado, porquê?
7. Se é registado, o modo como é registado satisfaz às necessidades do departamento?
8. Se não, como acha que deveria ser feito?
9. Como é feito o controlo de custos de manutenção?
10. Descreva todo o processo de manutenção de equipamento.

ANEXO B: Guião de Entrevista Semi-Estruturada



UNIVERSIDADE EDUARDO MONDLANE
FACULDADE DE CIENCIAS
DEPARTAMENTO DE MATEMÁTICA E INFORMÁTICA

GUIAO DE ENTREVISTA

Este guião de entrevista visa recolher informação relativa ao funcionamento e a estrutura orgânica e administração de equipamentos médicos do Departamento de Manutenção do Ministério de Saúde, no âmbito do trabalho de Licenciatura em Informática do estudante **Eurico José**, do Departamento de Matemática e Informática da Universidade Eduardo Mondlane, cujo o tema é:

Uso das Metodologias Orientadas a Objectos para Migração de Sistemas de Informação

Legados:

Estudo de Caso: Sistema de Gestão de Equipamentos Médicos.

A informação que for colhida será tratada com confidencia e somente para o fim a cima descrito.

Local: Departamento de Manutenção

Hora de início: ____:____

Data ____/____/____

Hora de fim : ____:____

Nome do Entrevistado: _____

Cargo / Função : _____

Repartição / Sector: _____

ASSUNTOS POR ABORDAR NA ENTREVISTA

1. Estrutura Organizacional e fluxo de Informação do MISAU e DM
2. Sistemas e Tecnologias de Informação do DM
3. Gestão de Equipamentos Médicos
4. Custos de Manutenção de Equipamentos

ANEXO C: Figuras Ilustrativas das Metodologias OO

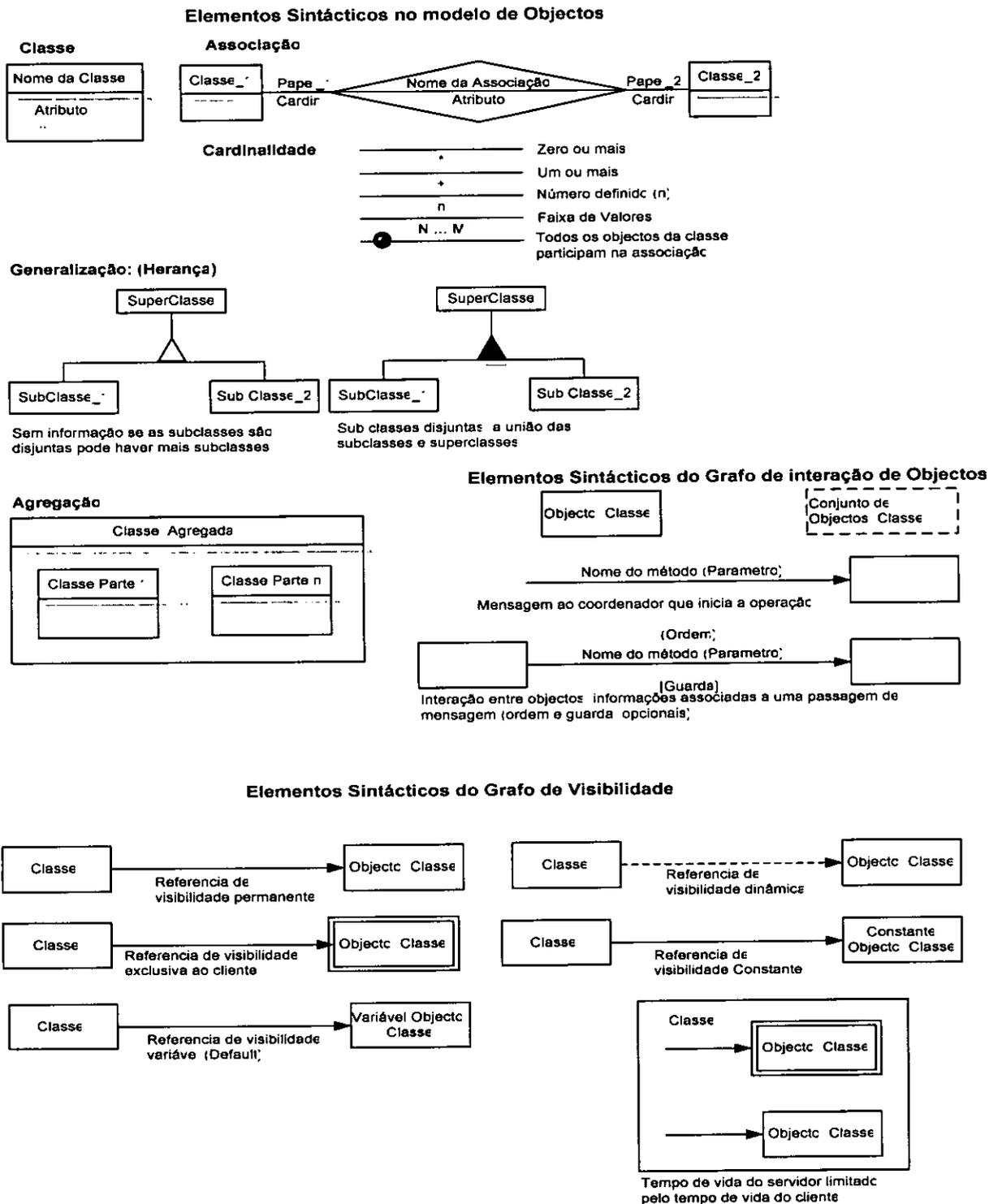
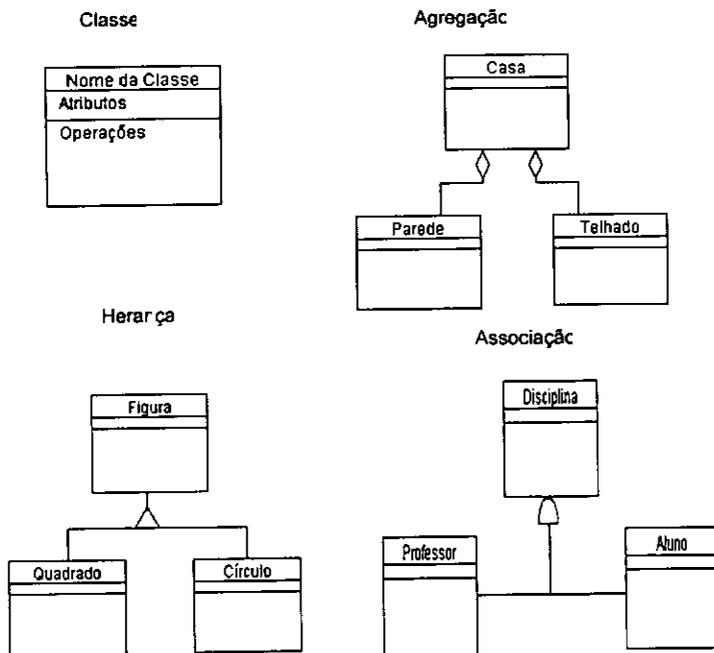


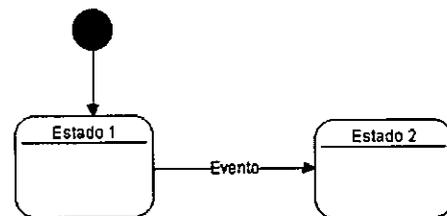
Figura C.1 Elementos Sintácticos de Análise e Desenho em FUSION

Elementos Sintácticos no Modelo de Objecto



Elementos Sintácticos no Modelo Dinâmico

Diagrama de Fluxo de Eventos



Elementos Sintácticos no Modelo Funcional

Diagrama de Fluxo de Dados

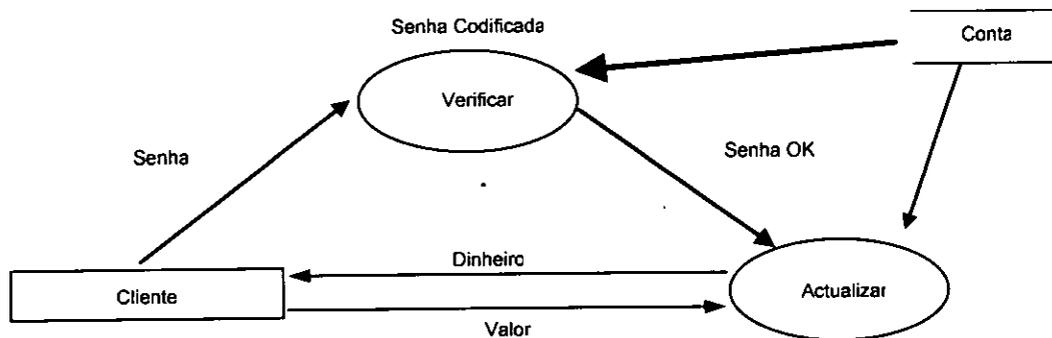
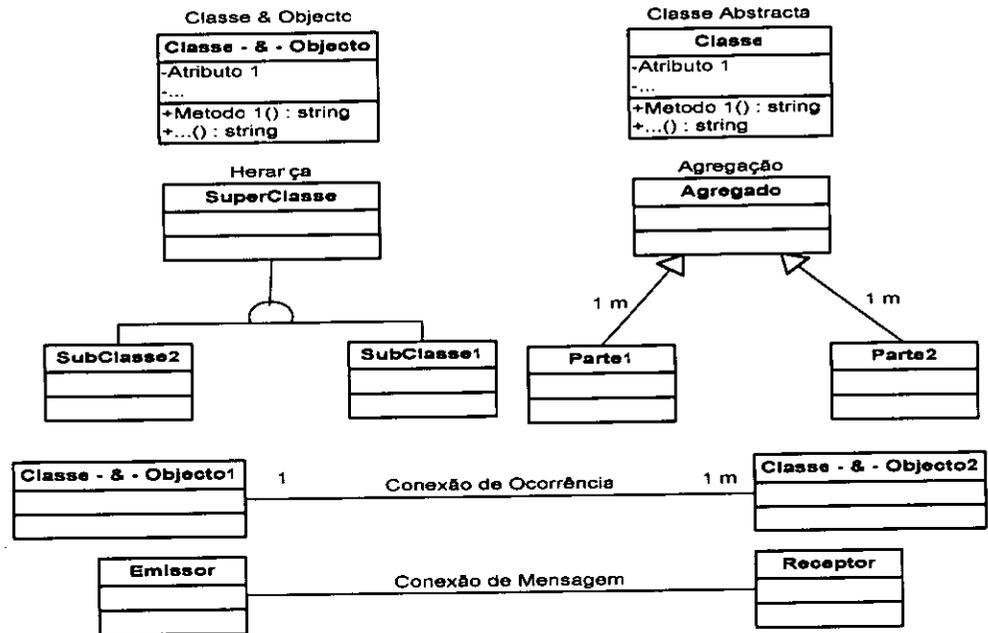
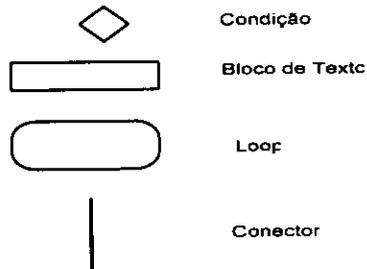


Figura C. 2 Elementos Sintácticos de Análise e Desenho em OMT

Elementos Sintáticos do Modelo de Objectos



Elementos Sintáticos do diagrama de Serviço



Elementos Sintáticos do diagrama de estado do Objectos

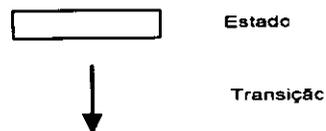


Figura C.3 Elementos Sintáticos de Análise e Desenho em COAD-YOURDON

ANEXO D: Modelo de Teste

D1. Teste Módulo – Inventariação de Equipamento

Descrição: Este módulo serve para o registo de todo tipo de equipamento médico existente numa US.
Este documento serve para testar as funcionalidades deste módulo.

Casos de Teste

1. Interface;
2. Estrutura (apresentação e organização de dados);
3. Funcionalidade;
4. Comparação com o sistema legado (Sim-Organizer);
5. Observações.

1. Interface

1.1. Como avalia a apresentação do formulário de registo de equipamento?

Muito Bom Bom Razoável Mau Não tenho opinião

1.2. Qual a sua avaliação em relação as cores usadas no formulário?

Muito Boas Boas Razoável Berrantes Não tenho opinião

2. Estrutura

2.1 Como avalia a estrutura de apresentação dos dados recolhidos no formulário?

Muito Boa Boa Razoável Não tenho opinião

2.2. Os dados recolhidos são suficientes? Sim Não

Se Não, quais os dados que faltam?

2.3. Há dados recolhidos desnecessários? Sim Não

Se Sim, indique-os _____

3. Funcionalidade

3.1. O módulo valida os dados introduzidos antes de os registar? Sim Não

Se Não, quais os dados que poderia validar e não valida. (Indique como deve ser feita a validação)

3.2. O módulo regista todos os dados introduzidos? Sim Não

Se Não, quais os dados que não são registados?

3.3. O módulo permite a visualização de dados registados? Sim Não

3.4. O módulo permite a visualização de dados de equipamento na forma de relatório?

Sim Não

Se Sim, este relatório está bem apresentado e visualiza dados correctos?

Sim Não Não tenho opinião

3.5. O módulo permite a alteração de dados de um equipamento anteriormente registado?

Sim Não

3.6. Como avalia a facilidade de utilização deste modulo?

Fácil Razoável Difícil Não tenho opinião

4. Comparação com o sistema legado (Sim-Organizer)

Se este módulo existe no sistema legado (Sim-Organizer) faça uma análise comparativa dos pontos 1 a 3 entre este modulo e o modulo do Sim-Organizer.

5. Observações Gerais.

D2. Teste Módulo – Abertura de FO

Descrição: Este módulo serve para abrir uma FO. A abertura de F.O. ocorre quando um técnico de manutenção pretende iniciar uma actividade de manutenção de equipamento. Neste caso para registar o início das actividades, deve abrir uma FO. Usar-se-á este módulo para abrir a F.O.

Este documento serve para testar as funcionalidades deste módulo.

Casos de Teste

1. Interface;
2. Estrutura (apresentação e organização de dados);
3. Funcionalidade;
4. Comparação com o sistema legado (Sim-Organizer);
5. Observações.

1. Interface

1.1. Como avalia a apresentação do formulário de abertura de F.O?

Muito Bom Bom Razoável Mau Não tenho opinião

1.2. Qual a avaliação que faz em relação às cores usadas no formulário?

Muito Boas Boas Razoável Berrantes Não tenho opinião

2. Estrutura

2.1 Como avalia a estrutura de apresentação dos dados recolhidos no formulário?

Muito Boa Boa Razoável Não tenho opinião

2.2. Os dados recolhidos são suficientes? Sim Não

Se Não, quais os dados que faltam? _____

2.3. Há dados desnecessários recolhidos? Sim Não

Se Sim, indique-os _____

3. Funcionalidade

3.1. O módulo valida os dados introduzidos antes de os registar? Sim Não

Se Não, quais os dados que poderia validar e não valida? (Indique como deve ser feita a validação)

3.2. O módulo regista todos os dados introduzidos? Sim Não

Se Não, quais os dados que não são registados? _____

3.3. O módulo permite a visualização de dados registados? Sim Não

3.4. O módulo permite a visualização de dados de abertura de FO na forma de relatório?

Sim Não

Se Sim, este relatório está bem apresentado e visualiza dados correctos?

Sim Não Não tenho opinião

3.5. O módulo permite alteração de dados de abertura de FO anteriormente registada?

Sim Não

3.6. Como avalia a facilidade de utilização deste módulo?

Fácil Razoável Difícil Não tenho opinião

4. Comparação com o sistema legado (SIM-ORGANIZER)

Se este modulo existe no sistema legado (SIM-ORGANIZER), faça uma análise comparativa dos pontos 1 a 3 entre este módulo e o módulo do SIM-ORGANIZER.

5. Observações Gerais.

D3. Teste Módulo – Fecho de FO

Descrição: Este modulo serve para fechar uma F.O. O fecho de F.O. ocorre quando um técnico de manutenção termina uma actividade de manutenção de equipamento. Neste caso para registar o fim das actividade deve fechar a F.O aberta. Usar-se-á este módulo para fechar a F.O.

Este documento serve para testar as funcionalidades deste módulo.

Casos de Teste

1. Interface;
2. Estrutura (apresentação e organização de dados);
3. Funcionalidade;
4. Comparação com o sistema legado (Sim-Organizer);
5. Observações.

1. Interface

1.1. Como avalia a apresentação do formulário de fecho de F.O?

Muito Bom Bom Razoável Mau Não tenho opinião

1.2. Qual a avaliação que faz em relação às cores usadas no formulário?

Muito Boas Boas Razoável Berrantes Não tenho opinião

2. Estrutura

2.1 Como avalia a estrutura de apresentação dos dados recolhidos no formulário?

Muito Boa Boa Razoável Não tenho opinião

2.2. Os dados recolhidos são suficientes? Sim Não

Se Não, quais os dados que faltam? _____

2.3. Há dados desnecessários recolhidos? Sim Não

Se Sim, indique-os _____

3. Funcionalidade

3.1. O módulo valida os dados introduzidos antes de os registar? Sim Não

Se Não, quais os dados que poderia validar e não valida? (Indique como deve ser feita a validação)

3.2. O módulo regista todos os dados introduzidos? Sim Não

Se Não, quais os dados que não são registados? _____

3.3. O módulo permite a visualização de dados registados? Sim Não

3.4. O módulo permite a visualização de dados de fecho de FO na forma de relatório?

Sim Não

Se Sim, este relatório está bem apresentado e visualiza dados correctos?

Sim Não Não tenho opinião

3.5. O módulo permite alteração de dados de fecho de FO anteriormente registada?

Sim Não

3.6. Como avalia a facilidade de utilização deste módulo?

Fácil Razoável Difícil Não tenho opinião

4. Comparação com o sistema legado (SIM-ORGANIZER)

Se este modulo existe no sistema legado (SIM-ORGANIZER), faça uma análise comparativa dos pontos 1 a 3 entre este modulo e o modulo do SIM-ORGANIZER.

5. Observações Gerais.

D4. Teste Módulo – Impressão de Relatórios

Descrição: Este módulo serve para imprimir diferentes relatórios (fichas e relatórios de gestão).
Este documento serve para testar as funcionalidades deste modulo.

Casos de Teste

1. Interface;
2. Funcionalidade;
3. Comparação com o sistema legado (Sim-Organizer);
4. Observações.

1. Interface

1.1. Este módulo é composto por diferentes formulários de parametrização de dados de entrada para impressão de diferentes relatórios e fichas. Como avalia a apresentação desses formulários?

Muito Bom Bom Razoável Mau Não tenho opinião

1.2. Qual a sua avaliação em relação às cores usadas nesses formulários?

Muito Boas Boas Razoável Berrantes Não tenho opinião

2. Funcionalidade

2.1. O módulo permite a visualização e a impressão de todas fichas e relatórios de gestão necessários?

Sim Não

Se Não, quais as fichas e os relatórios de gestão que o modulo não imprime? _____

2.2. Os relatórios impressos por este módulo contêm dados correctos?

Todos Alguns

Se Alguns, quais os relatórios contêm dados incorrectos? _____

2.3. Os relatórios impressos por este módulo estão bem apresentados?

Todos Alguns

Se alguns, quais os relatórios que não estão bem apresentados? _____

2.4. Como avalia a facilidade de utilização deste módulo?

Fácil Razoável Difícil Não tenho opinião

3. Comparação com o sistema legado (SIM-ORGANIZER)

Se este módulo existe no sistema legado (SIM-ORGANIZER), faça uma análise comparativa dos pontos 1 a 2 entre este módulo e o módulo do SIM-ORGANIZER.

4. Observações Gerais.

ANEXO E: Ecrãs de Comparação do Sistema Legado e do Novo

A Figura E.1 e Figura E 2 ilustram os formulários de inventariação de equipamento no sistema legado e novo sistema, respectivamente.

Data: 1998/04/21 **Executor:** Francisco Bazima

Localização: Gaza Xai-Xai Cidade Centro de saúde da cidade

Equipamento de Electromedicina: Diagnóstico

Unid. Sanitária: Centro de saúde da cidade **Nomenclatura:** Esterilizador, Vapor, Vertical

Identificação: MG-XAC-1-01-0022 **Marca:** XXXXX

Departamento: **Modelo:** DSA

Serviço: Esterilização **Nr série:** 138-D

Localização: (Xai-Xai Cidade - Centro de saúde da cidade - - Esterilização)

Observação geral: (Esterilizador, Vapor, Vertical - *** - DSA - 138-D)

Equipamento de Electromedicina: Diagnóstico

Utilizador respons.: Olivia **Telefone:**

Figura E.1 Formulário de Inventariação de Equipamento no Sistema Legado

Inventário de Equipamentos

Data de Inventariação: 20-04-2008

Província: Cidade de Maputo **Distrito:** DISTRITO URBANO Nº1

Unidade Sanitária: Posto de Saúde de Hulene

Localização **Especificação Técnica** **Documentação** **Acessórios Entregues** **Observação e Outros Dados**

Departamento: **Identificação:**

Serviço: Administração **Nº Património:**

Local: **Marca:** Adams equipment

Data Aquisição: 20-04-2008 **Modelo:**

Data Instalação: 20-04-2008 **Nº Série:**

Valor de Compra: MZM **Nomenclatura:**

Financiador: Cooperação Italiana **Área:** Radiologia

Designação: Cadeira estomatologica simples

Propri. do MISAU LEASING

Registar **Cancelar** **Fechar**

Figura E 2 Formulário de Inventariação de Equipamento no Novo Sistema

A Figura E.3 e a Figura E.4 ilustram os formulários de inventariação de US no sistema legado e novo sistema, respectivamente.

Unidades sanitárias

Número : MG-BIL-0-01

Unidade: Posto de Saúde de h

Distrito: [dropdown]

Província: Geza

Código: MG-BIL

Distrito: Bilene

Record: 1 of 12

Figura E.3 Formulário de Inventariação de Unidade Sanitária no Sistema Legado

Unidade Sanitária

Província: Cidade de Maputo

Distrito: DISTRITO URBANO Nº1

Data de Inventariação: 20-04-2008

Nome US: [input]

Identificação: [input]

Tipo: Centro de Formação

Especificação | Camas e Serviços | Electricidade, Água e Comunicação | Observação

Ano de Construção: [input]

Número de Edifícios: [input]

População Abrangida: [input]

Distância a U.S Ref. (KM): [input]

Distância a U.M Ref. (KM): [input]

Coordenadas GPS:

Norte: 00,00,00

Sul: 00,00,00

Este: 00,00,00

Oeste: 00,00,00

Nº Património: [input]

Nº SISD: [input]

Foto: [input]

Registrar | Cancelar | Fechar

Figura E.4 Formulário de Inventariação de Unidade Sanitária no Novo Sistema

A Figura E.5 ilustra o formulário de registo de requisição de intervenção no novo sistema. Esta funcionalidade não existe no sistema legado.

Requisição de Intervenção

Localização

Provincia: Cidade de Maputo | Distrito: DISTRITO URBANO Nº1

US: Posto de Saúde de Hulene

Departamento: | Serviço: Administração

Responsável: | Contacto:

Equipamentos

Código	Nº Série	Nomenclatura
--------	----------	--------------

Código: | **Nº Série**: | **Nome**: |

Descrição da Avaria:

Avaria de: Equipamento Serviço

Execução: Normal Urgente Imediata

Tipo Manutenção: Preventiva Correctiva

Contrato de Manutenção: Sim Não

Nº Contrato: | **Empresa**:

Registrar **Fechar**

Figura E.5 Formulário de Registo de Requisição de Intervenção

A Figura E.6 e a Figura E.7 ilustram os formulários de abertura e de Fecho de FO no sistema legado e no novo sistema, respectivamente.

Manutenção Preventiva:
 Instalação/Sem ID:

Folha Nr.: MG-MAN-2-01-0006

Informação geral | Inventário | Descrição avaria/instalação | Facturação

Preenchida por: Francisco Bazima
 Colocação: Electromedicina
 Pessoa a contactar: Carlos Sítio
 Telefone:

Data de entrada: 1998/03/06
 Data Início Trabalho: 1998/03/07
 Data Requisição Material:
 Data Entrega Material: →
 Data Fim Trabalho: 1998/03/07
 Data de saída: 1998/03/07

Figura E.6 Formulário de Abertura e Fecho F.O no Sistema Legado

Folha de Obra

Requisição

Número

Identificação

F.O. Número
 Data Abertura Horas
 Data Fecho Horas

Execução

U. Sanitária
 A Ser Executado Por Oficina

Equipamento/Serviço | Material Aplicado | Mão de Obra | Deslocação | Relatório Técnico | Fecho

Figura E.7 Formulário de Abertura e Fecho F.O no Novo Sistema

A Figura E.8 e a Figura E.9 ilustram o formulário de registo de funcionario no sistema legado e no novo sistema, respectivamente.

The screenshot shows a window titled "Lista Trabalhadores" with a table containing employee data. The table has three columns: "Responsável", "Categoria", and "Oficina". The records are as follows:

Responsável	Categoria	Oficina
Francisco Mucavele	Elementare	Pintura
Marcos Daniel Chiangul	Elementare	Electricidade
Marília Elsa Figueiredo Nhanturr	Basico	Refrigeração
Tomás Tovela	Elementare	Mecanica auto

At the bottom of the window, there is a record navigation bar showing "Record: 12 of 12" with navigation icons.

Figura E.8 Formulário de Registo de Funcionário no Sistema Legado

The screenshot shows a window titled "Funcionário" with a registration form. The form is titled "Dados do Funcionário" and contains the following fields:

- Nome: Text input field
- BI/Passaporte/Dire: Text input field
- Número: Text input field with a tooltip that reads "Número do Funcionário a nível do MISAU ou da Função Pública"
- NUIT: Text input field
- Nível de Escolaridade: Dropdown menu with "Sem Escolaridade" selected
- Oficina: Dropdown menu with "Bate Chapa" selected
- Categoria: Dropdown menu with "Bacharel" selected
- SPM: Dropdown menu with "Dept. Manutenção" selected

At the bottom of the form, there are three buttons: "Registrar", "Cancelar", and "Fechar".

Figura E.9 Formulário de Registo de Funcionário no Novo Sistema