

Sistema Pericial para auditoria de registos clínicos

Luisa L. Alexandre

Dissertação submetida ao Instituto Superior de Engenharia do Porto
para a obtenção do grau de Mestre em
Engenharia de Computação e Instrumentação Médica

Orientadores

DOUTORA GORETI MARREIROS
Professora Adjunta do Departamento de Informática
Instituto Superior de Engenharia do Porto

DOUTOR ALBERTO FREITAS
Professor Auxiliar da Faculdade de Medicina
Universidade do Porto

Porto, 24 de Novembro de 2013

A journey of a thousand miles begin with a single step.
Lao Tzu - Filósofo Chinês

Agradecimentos

Aos meus orientadores, Professora Doutora Goretí Marreiros e Professor Doutor Alberto Freitas, não só pela orientação do trabalho, como também pela disponibilidade, estímulo e confiança. Obrigada pelos vossos comentários, sugestões e indicações precisas e incisivas.

Ao Instituto Superior de Engenharia do Porto agradeço a oportunidade de realizar este trabalho, ao Departamento de Ciências da Informação e da Decisão em Saúde (CIDES) da Faculdade de Medicina da Universidade do Porto agradeço a disponibilização de meios e recursos necessários à realização do mesmo.

Agradeço a todos os meus colegas do CIDES pelas discussões científicas, ajuda e também pelo ambiente agradável que sempre me proporcionaram.

Agradeço também aos meus colegas do Mestrado em Engenharia de Computação e Instrumentação Médica pelo incentivo e ajuda, particularmente a Sara Cabral que além da amizade, me apoiou em todos momentos desta etapa.

À minha mãe e ao meu pai quero agradecer tudo o que fizeram e continuam a fazer por mim, apoiando-me sempre, e indicando o caminho correcto a seguir.

Ao Henrique Tavares obrigada pela paciência, carinho e pelas tentativas de demonstrações de interesse no meu trabalho.

E a todas as outras pessoas, que directa ou indirectamente colaboraram com o sucesso deste trabalho.

Resumo

O registo clínico é um documento hospitalar confidencial, nele é registado todo o percurso clínico de um determinado paciente. Esse percurso é registado através da atribuição de códigos descritos num sistema de classificação, também denominados de dados da codificação. Os dados da codificação são armazenados eletronicamente numa Base de Dados, e é a partir dela que certas medidas, tais como faturação hospitalar, financiamentos hospitalares anuais, tratamentos médicos, ou até mesmo dados epidemiológicos, são adotadas.

Portanto torna-se fundamental a garantia da qualidade na área da codificação clínica, para isso é necessário recorrer a processos de auditorias aos registos clínicos. O processo de auditoria é uma atividade independente de avaliação, cujo objetivo visa acrescentar valor e melhorar os objetos e processos da auditoria. Atualmente uma ferramenta denominada de Programa Auditor é utilizada, contudo essa ferramenta demonstra uma tecnologia já ultrapassada.

A Tese que se pretende defender é a de que, através de Sistemas Periciais, é possível realizar auditorias internas aos registos clínicos. Pretende-se ainda demonstrar que dos Sistemas Periciais é um benefício para os peritos na área, pois diminui a probabilidade de erros e torna o processo de verificação menos moroso.

Neste contexto, o objetivo desta Dissertação prende-se em definir e implementar um Sistema Pericial que permita a auditoria de registos clínicos hospitalares. O Sistema Pericial deverá ainda ser capaz de traduzir o raciocínio do perito, detetando assim diversos tipos de erros que traduzem num registo clínico não conforme.

Por sua vez, foi desenvolvido um protótipo de um Sistema Pericial para auditoria de registos clínicos em linguagem PROLOG. Este mesmo protótipo serviu de base à realização de uma experiência que permitiu comparar com o programa Auditor e verificar a sua aplicabilidade no dia-a-dia hospitalar.

Abstract

The medical record is an hospital confidential document, that is recorded throughout the clinical course of a patient. This clinical course is registered by assigning codes described in a classification system. The data are stored electronically in a database, and it is from this data that certain measures like hospital billing, hospital annual funding, medical treatments, or even epidemiological data, are taken.

Therefore it is essential the assurance of quality in the area of clinical coding, and for this it is necessary to use audit processes into clinical records. The audit process is an independent evaluation activity whose objective is to add value and improve the objects and processes of audit. At the present a tool called Program Auditor is used, however this tool demonstrates an outdated technology.

This thesis demonstrate the use of expert systems, to possibility and assist the internal audits medical records. It is also intended to demonstrate that the use of new technologies is a benefit to the field experts, because it reduces the likelihood of errors and makes the verification process less slow.

In this context, the aim is define and implement an expert system that allows auditing of hospital clinical records. The expert system should be able to translate the experts thinking, to detects various types of errors that translate into a clinical record non-conforming.

Meanwhile was developed a prototype of an expert system for auditing medical records in PROLOG. This same prototype served as a basis for making an experience that allowed us to compare tools and draw some conclusions about the work.

Conteúdo

Agradecimentos	vi
Resumo	viii
Abstract	x
Conteúdo	xii
Lista de Figuras	xiii
Lista de Tabelas	xv
Acrónimos	xvii
1. Introdução	1
1.1 Enquadramento	1
1.2 Apresentação do trabalho	2
1.3 Motivação	3
1.4 Organização da Dissertação	3
2. Codificação Clínica	5
2.1 Contextualização	5
2.2 Classificação Internacional de Doenças (CID-9-MC)	6
2.3 Grupos de Diagnóstico Homogéneos (GDH)	7
2.4 Aplicação da codificação clínica	10
2.4.1 Utilização da codificação em Portugal e no resto do Mundo	10
2.4.2 Controlo da codificação	11
3. Auditoria	15
3.1 Contextualização	15
3.2 Ferramentas de Auditoria	16
3.3 Utilização de Sistemas Periciais na área médica	20

4. Desenvolvimento do Sistema Pericial	25
4.1 Definição do domínio do problema	25
4.2 Selecção do perito	28
4.3 Desenho Conceptual e estudo da viabilidade técnica	29
4.4 Selecção de Hardware e Software	30
4.5 Aquisição do conhecimento e representação do Conhecimento	31
4.6 Construção de um protótipo	32
4.6.1 Construção da Base de Dados Local	34
4.6.2 Construção de uma aplicação para o tratamento de dados	34
4.6.3 Construção da conexão à Base de Dados Local	37
4.6.4 Construção das regras	39
4.6.5 Construção da acção	45
4.7 Validação e Verificação do sistema	46
4.8 Aceitação do sistema pelo utilizador	48
5. Considerações Finais	49
5.1 Conclusões	49
5.2 Limitações e Trabalho futuro	50
Bibliografia	52
A. Tabela de regras técnicas e clínicas	61
B. Ficheiro Output do Programa Auditor	65
C. Ficheiro Output do Sistema Pericial	67

Lista de Figuras

2.1	Esquema de partição de uma GCD em GDH	9
3.1	Menus de trabalho do programa auditor	19
3.2	Arquitetura de um Sistema Pericial	21
4.1	Modelo de desenvolvimento de ESAQAS	26
4.2	Fluxo do registo clínico no Hospital	27
4.3	Arquitetura da aplicação desenvolvida	30
4.4	Transcrição do raciocínio da validação técnica da proveniência do paciente de outro hospital.	33
4.5	Modelo de dados do Sistema Pericial baseado nos ficheiros de exportação	35
4.6	Arquitectura da aplicação para tratamento de dados	36
4.7	Código com função específica para leitura do ficheiro CSV	37
4.8	Queries de inserção na Base de dados.	38
4.9	Extrato de código PROLOG com comandos para inicializar e encerrar a ligação a base de dados	38
4.10	Predicados de associação aos campos da BD	39
4.11	Predicados de associação e preenchimento automático de etiquetas . .	41
4.12	Extracto de código FLEX com regras técnicas	43
4.13	Extrato de código PROLOG referente ao procedimento auxiliar de verificação da data e código FLEX de validação na regra	43
4.14	Extrato de código PROLOG criado para agrupar códigos do mesmo registo clínico	44
4.15	Extrato de código FLEX de validação de regras clínicas referentes a coexistência de códigos de episiotomia	45
4.16	Conexão entre ficheiros do Sistema Pericial	46

Lista de Tabelas

4.1	Resultados clínicos da verificação do sistema comparativamente ao Programa Auditor	47
-----	--	----

Acrónimos

- ACSS - Administração Central do Sistema de Saúde, I.P
- AMACC - Associação dos Médicos Auditores e Codificadores Clínicos
- AP-DRG - *All Patient Diagnosis Related Group*
- BD - Base de dados
- ChronoMedIt - *Cronological Medical Audit*
- CID - Classificação Internacional de Doenças
- CID-9 - Nona edição da Classificação Internacional de Doenças
- CID-9-MC - Nona edição da Classificação Internacional de Doenças, Modificação Clínica
- CID-10 - Décima edição da Classificação Internacional de Doenças
- CID-11 - Décima primeira edição da Classificação Internacional de Doenças
- CIDES - Departamento de Ciências da Informação e Decisão em Saúde, FMUP
- CSV - *Comma Separated Values*
- DBMS - *DataBase Management Systems*
- DLL - *Dynamic-link library*
- EUA - Estados Unidos da América
- FMUP - Faculdade de Medicina da Universidade do Porto
- GC - Gabinete de Codificação
- GCD - Grandes Categorias Diagnósticas
- GDH - Grupos de Diagnóstico Homogéneos
- GECAD - Grupo de investigação em Engenharia do Conhecimento e Apoio à Decisão

- ICM - Índice de case-mix
- IGIF - Instituto de Gestão Informática e Financeira
- IPAI - Instituto Português de Auditoria Interna
- LPA - *Logic Programming Associates*
- MC - Modificação Clínica
- NHIN - *National Health Information Network*
- ODBC - *Open DataBase Connectivity*
- OMS - Organização Mundial de Saúde
- PHP - *PHP: Hypertext Preprocessor*
- PROLOG - *PROgrammation LOGique*
- R/N - Recém Nascido
- RICE - *Risk Identification and Control Evaluation*
- SANDS - *Servise-oriented Architecture for Clinical Decision Support*
- SGISS - Sistema de Informação para a Gestão dos Serviços de Saúde
- SNS - Serviço Nacional de Saúde
- SP - Sistema Pericial
- SQL - *Structured Query Language*
- SWI - *Sociaal-Wetenschappelijke Informatica*

Introdução

Neste capítulo descreve-se sumariamente o problema em estudo no âmbito do desenvolvimento de um Sistema Pericial para auditoria na área médica, assim como a motivação para a realização do mesmo, e os objetivos definidos. Por último, é apresentada a estrutura da presente dissertação.

1.1 Enquadramento

Um conceito genérico de auditoria fornecido pelo IPAI (Instituto Português de Auditoria Interna) diz que "a auditoria interna é uma atividade independente, de avaliação objetiva e de consultadoria, destinada a acrescentar valor e melhorar as operações de uma organização" [1]. Pode-se assim referir que uma auditoria da atividade médica, deve ser vista de forma semelhante, ou seja é uma auditoria de qualquer atividade, que ocorra dentro do serviço de saúde e que possa ter uma consequência direta ou indireta, no paciente.

Todavia existem outras definições que relacionam o conceito de auditoria com as diferentes áreas médicas. Uma forma de as apresentar é subdividindo a área médica em 3 subcamadas. A primeira subcamada mais específica, denomina-se auditoria médica, e é a revisão de todos os processos relacionados com o atendimento clínico prestado aos pacientes e realizados somente pela equipa médica. A segunda é relacionada com a auditoria clínica e efetua a revisão de todos os processos relacionados com atos clínicos, ou seja, atividades relacionadas com o tratamento clínico fornecido ao paciente por todos os profissionais de saúde (exemplo: médicos e enfermeiros). A última camada é a mais generalista e refere a auditoria do atendimento ao paciente, ou seja, é a revisão de toda a atividade realizada dentro dos serviços de saúde ao

longo do percurso realizado pelo paciente [2].

Desde os primórdios que os médicos examinam os efeitos dos seus tratamentos e conseqüentemente avaliam os seus resultados, praticando uma auto-auditoria sobre as suas decisões, e alterando, se necessário, os seus métodos. Essa retroalimentação contínua conduz a uma significativa melhoria na prestação do serviço. Logo, a prática da auditoria na área médica não é um novo conceito, no entanto, e por forma a homogeneizar essa prática, com o passar dos anos foram criadas diretrizes, tornando assim a auditoria numa ferramenta de avaliação segura e eficaz.

A correta utilização da auditoria na área médica pode resultar em grandes benefícios tanto para os pacientes, como para os profissionais de saúde, pois assegura uma contínua qualidade do serviço e por consequência uma fiscalização dos custos com recursos.

1.2 Apresentação do trabalho

Este trabalho insere-se no Grupo de Qualidade de dados pertencente ao departamento de Ciências da Informação e da Decisão em Saúde (CIDES) da Faculdade de Medicina da Universidade do Porto. A finalidade/objetivo deste projeto prende-se com a definição e implementação de um Sistema Pericial (SP) capaz de realizar a auditoria de registos clínicos hospitalares pertencentes a uma base de dados clínico-administrativa, também denominada de WebGDH. Essa implementação focar-se-á em todos os dados administrativos e em dados clínicos específicos da área da obstetrícia.

O SP deverá ser capaz de traduzir o raciocínio do perito, ou seja, do médico auditor, e detetar/lidar com episódios hospitalares não conformes. São considerados os seguintes tipos de erro:

- Erros associados ao diagnóstico principal: diagnóstico principal em branco, diagnóstico principal inválido, ou seja, com 4º ou 5º dígito em falta, ou diagnóstico principal inaceitável clinicamente, ou seja, não correspondente ao registo clínico;
- Erros associados ao procedimento cirúrgico de codificação obrigatória: procedimentos incoerentes clinicamente, procedimentos inválidos, ou procedimentos duplicados;
- Erros associados aos diagnósticos adicionais: diagnóstico adicional inaceitável

e/ou inespecíficos;

- Erros associados à codificação das causas externas: códigos de causa externa errado, ou seja com o 4º dígito em falta ou inválido;
- Erros associados aos dados administrativos: Dados em falta ou inválidos, proveniência inválida ou desconhecida, erros associados a datas de nascimento/admissão/cirurgia/alta, erros associados ao peso à nascença, entre outros dados administrativos.

1.3 Motivação

Numa auditoria realizada em 2010 pela Administração Central do Sistema de Saúde (ACSS) foi verificado que de entre 1061 episódios auditados, aproximadamente 66% estavam não conformes, ou seja, que não respeitavam as normas de codificação clínica, ou que de alguma forma apresentavam erros ao nível dos dados administrativos [3].

Esses erros implicam inevitavelmente uma avaliação do desempenho hospitalar e financiamento incorretos, implicando assim maiores gastos para o Serviço Nacional de Saúde (SNS). Atualmente, e com os consequentes cortes nas verbas disponíveis para a saúde, é cada vez mais importante e essencial garantir um financiamento justo e adequado à faturação anual de cada hospital.

Por outro lado, a informação corretamente codificada permite recolher informações íntegras, que se traduzem, posteriormente, em indicadores de morbilidade exatos e possibilitando também conhecer as necessidades de saúde da população portuguesa.

Nesse contexto, há um crescente interesse por parte do Ministério da Saúde, mais especificamente da ACSS, em ferramentas que possibilitem a averiguação e melhoria da qualidade dos dados, nesse caso, através da auditoria de registos clínicos codificados.

1.4 Organização da Dissertação

Esta dissertação está organizada em 5 capítulos principais; o presente capítulo contextualiza o trabalho, aludindo a importância de uma ferramenta de auditoria na área da codificação clínica. Também é descrito brevemente o âmbito do trabalho, os contributos e os respetivos objetivos.

No Capítulo 2 é feita uma abordagem teórica à codificação clínica e as áreas relacionadas, como também a aplicação da codificação em Portugal e o respetivo controlo da mesma.

No Capítulo 3 é abordado a área da Auditoria, fornecendo uma ligeira consideração histórica sobre conceito, e as suas ferramentas. Por fim procurou-se abordar o programa Auditor, programa que atualmente está implementado na maioria dos hospitais pertencentes ao Serviço Nacional de Saúde, fornecendo algumas das suas características como também limitações.

O capítulo 4 expõe o desenvolvimento do Sistema Pericial para registo de auditorias clínicas baseado numa *guideline* denominada ESAQAS. Essa *guideline* conta com 8 fases para o desenvolvimento de um Sistema Pericial no âmbito da qualidade, que são ainda, nesse capítulo abordadas.

Por fim, no capítulo 5, tecem-se algumas considerações globais sobre o trabalho assim como sugestões de melhoria.

Codificação Clínica

Neste capítulo é apresentada uma pesquisa bibliográfica no âmbito da codificação clínica, de modo a que o leitor consiga contextualizar o domínio da presente dissertação. Posteriormente será demonstrado como é aplicada a codificação clínica em Portugal e no resto do Mundo e como se realiza o seu controlo.

2.1 Contextualização

Não é fácil criar uma visão uniforme da qualidade em saúde [4]. Cada interveniente do sistema possui uma visão diferente do conceito de qualidade. Para a indústria farmacêutica a qualidade advém dos medicamentos, enquanto para os médicos provém de melhores recursos e valorização das carreiras, já para os doentes a qualidade passa pela prestação de um cuidado de saúde eficiente, e para o Ministério da Saúde a qualidade provém de um sistema de saúde uniforme [5].

No Decreto-lei DL11/93 de 15 de Janeiro refere que "a incessante preocupação de propiciar aos utentes cuidados compreensivos e de elevada qualidade aconselha alterações de vulto na orgânica do SNS". Uma dessas alterações é a inserção da codificação clínica em Portugal que, além de ser pioneira na Comunidade Europeia, trouxe vantagens na gestão da doença e tratamento e na qualidade em saúde [6].

Inerentes à codificação, outras ferramentas também foram introduzidas, tal como as ferramentas de classificação de doenças, agrupadores de Grupos de Diagnóstico Homogêneos(GDH) e indicadores de qualidade [7].

2.2 Classificação Internacional de Doenças (CID-9-MC)

Em 1900, a Organização Mundial de Saúde (OMS) adotou pela primeira vez um método de Classificação Internacional de Doenças (CID). Esse método tinha como objetivo controlar e comparar as estatísticas das condições de morbidade e mortalidade, e possibilitar a indexação dos registos hospitalares para posterior armazenamento e/ou pesquisa. Revisões subsequentes à primeira versão aconteceram por forma a englobar e descrever novos conhecimentos na área e especificar cada vez mais os conhecimentos já adquiridos [8].

A nona edição da Classificação Internacional de Doenças (CID-9) foi publicada pela OMS em 1975. Essa edição foi imediatamente considerada como inadequada para a realidade do serviço de saúde nos Estados Unidos, assim, ainda em 1975 foi publicado uma modificação clínica (MC) do CID-9. O CID-9-MC possui como base a versão oficial da CID-9 e contém mais detalhes em áreas específicas, através da introdução de subclassificações de 5 dígitos nas subcategorias de 4 dígitos, já existentes nas versões anteriores [9, 10]. Atualmente, a sua publicação é composta por 3 volumes compilados no mesmo livro. No volume 1 e 2 encontram-se, respetivamente, a lista tabular e o índice alfabético das doenças e lesões traumáticas. No volume 3 encontra-se a lista tabular e o índice alfabético dos procedimentos. No total o CID-9-MC conta com aproximadamente 17000 códigos de diagnósticos distribuídos por 19 capítulos, sendo que destes, 17 são de diagnósticos, e os restantes são de classificações complementares [11].

Entretanto outras revisões foram posteriormente publicadas, como é o caso da CID-10, em 1992. As principais atualizações focalizavam-se na disponibilidade de mais códigos, com a inserção do 6º e 7º dígito, que possibilita a presença de 68000 códigos; os códigos da ICD-10, ao contrário da versão anterior, são obrigatoriamente alfanuméricos; e a lista tabular das doenças contém 21 capítulos, comparativamente com os 19 que a CID-9-MC possui [8, 12]. Mais recentemente, em Abril de 2007, foi lançado a 11º revisão da classificação internacional de doenças (CID-11) [13].

A utilização da codificação através da CID-9-MC, possui diversas vantagens, face às versões anteriores, em áreas distintas. Hazlewood em 2003 descreveu algumas dessas finalidades vantajosas, tais como: ser um método de controlo da qualidade; segurança e eficácia do tratamento médico; possibilidade de controlar a utilização de recursos e consequentemente melhorar a performance financeira e administrativa; fornecer equidade dos financiamentos anuais; possibilitar o estudo dos custos e/ou

resultados das várias opções de tratamento; planejar necessidades futuras atempadamente; reconhecer e identificar práticas de reembolso abusivas e/ou fraudulentas [14].

2.3 Grupos de Diagnóstico Homogéneos (GDH)

Os Grupos de Diagnóstico Homogéneos (GDH) são um sistema de classificação de doentes, que permite definir operacionalmente um conjunto de bens e/ou serviços que cada doente recebe aquando de um internamento hospitalar [15]. Este sistema agrupa os doentes através de uma vertente financeira, ou seja, correlaciona os tipos de doentes (informações demográficas) com os recursos consumidos ao longo de um internamento, criando assim grupos de pacientes que são uniformes tanto do ponto de vista clínico quanto a nível de consumo de recursos [16].

Os GDH foram desenvolvidos no término dos anos 60, na Universidade de Yale, nos EUA, por uma equipa multidisciplinar liderada por Robert B. Fetter e John Thompson. Essa equipa tinha como finalidade criar um método que possibilitaria a definição de produtos hospitalares, facilitando assim, a gestão e planeamento hospitalar através do controlo da utilização de recursos e medição da avaliação hospitalar [17]. Esse estudo foi motivado pela grande procura de informação nessa área que, posteriormente à implementação do programa Medicare. Este programa obrigou a todos os hospitais pertencentes ao programa Medicare, a possuírem uma ferramenta de classificação e revisão do produto hospitalar [18].

Durante essa investigação, foi desenvolvida então uma metodologia de classificação denominada de Grande Categoria de Diagnóstico ou GCD. Essa metodologia possibilita maior coerência na atribuição de um GDH. Assim, os GDH exigem a recolha de um conjunto mínimo de dados, como por exemplo, diagnóstico principal, outros diagnósticos, procedimentos, sexo, idade, destino após alta, e peso à nascença (caso seja um Recém-nascido) [15]. A primeira etapa para a classificação de um GDH é a divisão de um determinado episódio numa GCD. Cada GCD deve englobar diagnósticos que possuam uma homogeneidade ao nível anatómico, da classificação fisiopatológica ou da forma como os doentes são tratados clinicamente, originando assim GCD específicas, como é o exemplo da GCD 1 equivalente às doenças e perturbação do sistema nervoso. Posteriormente cada GCD é subdividida em GDH de acordo com a utilização de determinados procedimentos, idade, sexo, entre outras variáveis [18]. A metodologia da GCD ainda é atualmente utilizada na codificação. A título exemplificativo, a figura 2.1 apresenta a subdivisão do GCD 10, pertenc-

cente ao grupo das doenças e perturbações endócrinas, nutricionais e metabólicas, em respetivos GDH.

Muitas questões foram levantadas, aquando da implementação dos GDH, como o facto da não equidade dos dados e validação clínica. Esses problemas foram posteriormente eliminados através da utilização de uma linguagem comum para agrupar em GDH, a CID-9-MC [18]. Desde a sua primeira implementação até à atualidade os GDH sofreram diversas atualizações, de modo a que os grupos sejam cada vez mais homogéneos em relação ao consumo de recursos [17]. Atualmente e tal como descrito na Portaria nº 567/2006 de 12 de Junho, o agrupador GDH (sistema informático que contém as *guidelines* necessárias para agrupar os registos) utilizado em Portugal é o *"All Patient Diagnosis Related Group"* (AP-DRG).

As principais características da utilização dos GDH comparativamente a outros sistemas de classificação são [19]:

- Utilização dos GDH como meio de produção ou faturação:

Um modelo de financiamento é um conjunto de regras, rigorosamente delineadas, destinadas à subsistência dos recursos necessários para um cuidado de qualidade [20]. O modelo de financiamento baseado em GDH possui duas modalidades distintas. A primeira, caso seja o Estado a entidade financiadora (o que acontece maioritariamente em Portugal), assume a forma de um orçamento global que diferencia os valores consoante os GDH agrupados em cada hospital. A segunda modalidade assume como entidade financiadora os subsistemas de saúde ou terceiros, e o modo de faturação corresponde a pagamentos por episódios de internamento a preços médios dos GDH. Esses modelos de financiamento são divulgados e ajustados anualmente, através da Portaria Ministerial divulgada em Diária da República [21]. Para efeitos de faturação, e de modo a diferenciar os preços, cada GDH possui um peso relativo, esse peso é um coeficiente de ponderação que reflete o custo esperado de um determinado tratamento. A Portaria nº 132/2009 de 30 de Janeiro relata o preço base de um coeficiente de ponderação médio (igual a 1) em 2.396,24 Euros [22].

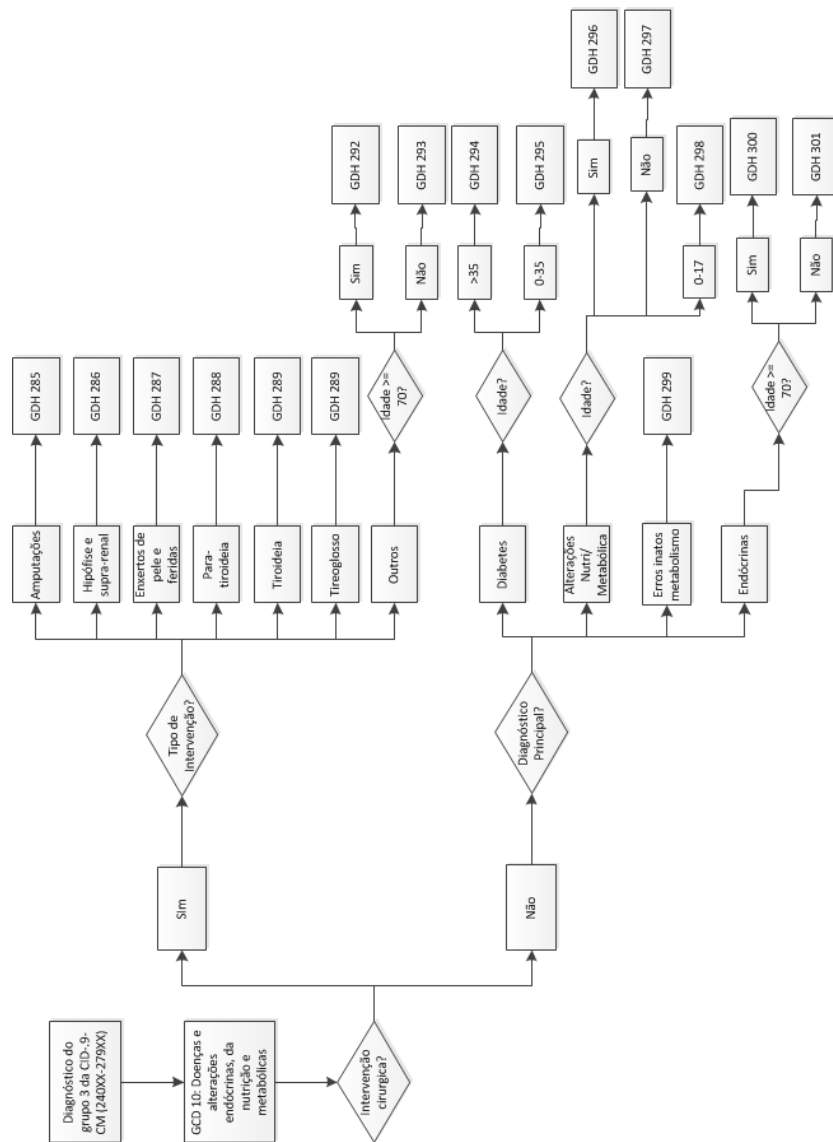


Fig. 2.1: Esquema de partição de uma GCD em GDH. Adaptado de [17].

- Possibilidade de medição do índice de case-mix hospitalar:

O índice de case-mix (ICM) é o grau ponderado da produção de um determinado hospital. Através do ICM é possível verificar a maior ou menor proporção de doentes tratados e a consequente complexidade dessas patologias [23]. A determinação do ICM é realizada através do rácio entre o número de doentes equivalente ponderados, pelos pesos relativos dos respetivos GDH e o número total de doentes equivalentes [24].

O ICM é utilizado para diferenciar os diversos hospitais, como também, tal como previsto nos contrato-programa, para calcular o financiamento anual dos hospitais [25, 26]. Através dos relatórios de contas disponíveis no Portal da Saúde é possível verificar que para o ano de 2008 (último ano publicado) o Hospital S. João, EPE possui um ICM de 1,5066 enquanto o Hospital S. Sebastião, EPE possui um ICM de 0,95.

2.4 Aplicação da codificação clínica

A codificação clínica traduz-se na tarefa de atribuição de códigos de um sistema de classificação a um registo clínico [7]. Raquel Braga, afirma que a codificação clínica permite harmonizar e categorizar os dados em códigos, sendo que esses dados são personalizados e traduzem todo o percurso clínico de um doente [27].

A aplicação da codificação é relativamente recente, e levou a extremas mudanças, contudo e tal como qualquer área que reflete a utilização de tecnologias, possui várias vantagens inerentes à sua implementação.

2.4.1 Utilização da codificação em Portugal e no resto do Mundo

A codificação nos EUA começou a ser implementada nos anos 70, os benefícios imediatos foram notórios, tais como a possibilidade de ajustar taxas de financiamento, ou a realização de uma documentação homogénea que por sua vez era uma mais valia para estudos clínicos e de performance, ou até mesmo a melhoria da qualidade do serviço prestado [26].

Após essa expansão nos EUA, a codificação difundiu-se e a sua utilização a nível internacional cresceu. Contudo, nem todos os países abordavam a codificação da mesma forma. A principal distinção era na aplicação do índice de case-mix

para financiamentos hospitalares, França, Itália, Irlanda, entre outros países, são exemplos da não utilização desse índice. Assim, é possível verificar que a ampliação da codificação foi grande. Contudo muitas foram as adaptações realizadas em cada país de modo a que fosse adequada a realidade financeira e tecnológica de cada um.

Roger France concluiu no seu estudo comparativo da aplicação da codificação em 25 países, que a falta de uniformidade na codificação de procedimentos (diferentes versões da CID) e nos agrupadores (diversas versões do GDH), tornam as comparações internacionais bastante difíceis [28].

Especificamente em Portugal, a codificação começou a ser implementada em 1990. Anteriormente a esse ano, muitas modificações foram tidas em conta, exemplos dessas modificações são [29]:

- Adaptação da base de dados dos GDH: Apesar de os hospitais já possuírem base de dados de registos clínicos, foi necessário a adaptação desta, por forma a possibilitar a inclusão dos GDH. Essa modificação possibilitou uma melhoria da performance clínica, da qualidade dos dados, e a eventual utilização dos mesmos para estudos.
- Formação de médicos codificadores: A formação de médicos codificadores que pudessem trabalhar com a CID-9-MC foi essencial para a unificação da codificação. Assim, foi possível que os registos clínicos fossem traduzidos para os códigos da CID-9-MC, armazenados nas bases de dados dos GDH e posteriormente utilizados para cálculos de faturação e financiamento.

De acordo com o referido anteriormente, pode-se constatar que com a implementação da codificação em Portugal, foi possível um grande avanço tecnológico, técnico, estrutural e de recursos, além de uma uniformidade da codificação a nível nacional.

2.4.2 Controlo da codificação

Os dados da codificação são armazenados eletronicamente numa base de dados (BD). Essas BD armazenam uma grande quantidade de dados clínicos, que são posteriormente utilizados pelos médicos de modo a serem adotadas medidas no tratamento de um determinado doente, e dados demográficos, que são mais de carácter administrativo e são utilizados para faturação hospitalar [30].

A base de dados utilizada em Portugal é orientada aos GDH e está presente em aproximadamente 95 hospitais do Serviço Nacional de Saúde (SNS). Por sua vez,

cada BD dos hospitais é centralizada numa única BD na Administração Central do Sistema de Saúde (ACSS), podendo ser utilizada para análises estatísticas e estudos epidemiológicos [29]. Como já foi referido, a codificação influencia a avaliação hospitalar e o seu financiamento anual. Assim, os erros ou determinadas alterações nas BD irão afetar essas duas medidas.

A qualidade dos dados é um conceito relativo, porque os dados podem ser considerados apropriados para um determinado propósito e inadequados para outro, ou seja a qualidade depende quase inteiramente no seu uso/aplicação [24]. Assim um dado presente na BD pode ser apropriado para o uso clínico e desapropriado para a codificação. Num estudo, que ocorreu em 2010, Costa [31] demonstrou alguns erros e/ou inconsistências em dados presentes na BD hospitalar entre os períodos de 2000 e 2007. De entre os dados, foram encontrados: erros de não preenchimento do campo "tipo de episódio", tempos de internamento com valores negativos, erros no campo "idade", incongruências nas transferências hospitalares, inconsistências no campo "número de episódio", registos duplicados, entre outros erros que demonstram a instabilidade dos dados e por sua vez a má produção de indicadores e financiamento hospitalares [31].

Outro estudo de Costa, realizado no mesmo trabalho, demonstrou a influência dos erros referidos anteriormente no financiamento hospitalar. O estudo abrangia 16 hospitais e como resultado obteve dados que demonstravam o prejuízo/benefício desses hospitais. No caso do Hospital A, e devido à quantidade de erros na sua BD, o financiamento atingiu uma redução de 4,05% que corresponde num total de menos 245 917,65 euros por ano [31].

Noutra perspectiva, a manipulação deliberada dos dados de modo a aumentar o financiamento hospitalar também é uma realidade. Steinwald demonstra no seu estudo que com a iniciação da aplicação da codificação clínica e do modelo de financiamento por GDH, ocorreu simultaneamente um aumento significativo nas tecnologias utilizadas, e na codificação de mais informações clínicas como por exemplo diagnósticos secundários e procedimentos [32].

Outro estudo realizado por Dismuke em 1999, tem como objetivo demonstrar que os administradores reagiram de forma positiva ao sistema de financiamento por GDH, ou seja, através da utilização de tecnologias de diagnóstico como tomografia axial computadorizada (TAC), eletroencefalograma (EEG) e ecografias (ECO), o hospital aumenta a sua eficiência e produtividade e por sua vez aumenta o financiamento anual do hospital. Dismuke tornou visível com o seu estudo que a utilização de TAC, que é uma tecnologia mais cara, aumentou significativamente após a apli-

cação do novo método de financiamento enquanto que a utilização de ECO, que é uma tecnologia similar contudo mais barata, diminuiu [20]. É de realçar que o uso de uma tecnologia mais cara implicou um aumento na classificação do GDH, que por sua vez influencia o aumento do índice de case-mix.

Através do referido anteriormente, é possível concluir que uma avaliação/controlo da codificação clínica terá um impacto positivo tanto a nível hospitalar, pois usufruirá de uma correta faturação, como ao nível do SNS que saberá o verdadeiro gasto dos hospitais sendo assim capaz de determinar o valor adequado de financiamento.

Portanto, e na tentativa de garantir uma codificação de qualidade, foram tomadas diversas ações. A primeira levada a cabo em 1988, foi a formação de médicos codificadores, por forma a minimizar os erros da codificação. A segunda ocorreu em 1995 e incentivava a realização de auditorias externas à codificação clínica nos hospitais. E por fim em 1996 foi realizado o primeiro curso de médicos auditores da codificação clínica, de maneira a ser possível a realização de auditorias internas à codificação [19].

Auditoria

Neste capítulo será realizada uma breve descrição da auditoria e o seu conceito histórico e focalizar-se-á nas suas ferramentas. Abordará também um exemplo prático que atualmente é utilizado em Portugal - o programa Auditor. Também neste capítulo serão referidos as vantagens da utilização da Inteligência Artificial na contribuição de uma ferramenta de auditoria. Serão ainda descritos alguns sistemas existentes, sendo que alguns apresentam um carácter generalista e outros são dedicados à área médica.

3.1 Contextualização

Nos meados do século XVI, na Inglaterra, foi criado um imposto sobre as mercadorias, esse imposto, era baseado nos lucros da empresa. De modo a evitar fraudes a esse imposto e também como método de revisão de contas, surgiu o termo "auditoria" [33]. Hoje em dia a auditoria possui uma aplicação mais vasta e o seu conceito passou a estar associado com qualquer apreciação de procedimentos ou operações com vista à obtenção da conformidade dos mesmos [34].

De acordo com, a Norma Portuguesa EN ISSO 19011/2003, "a auditoria é um processo sistemático, independente e documentado, para obter registos, afirmações factuais ou outra informação que seja relevante e a respetiva avaliação objetiva com vista a determinar em que medida os critérios de auditoria são satisfeitos".

O termo auditoria também está bastante relacionado com o termo qualidade, pois a realização de auditorias de forma contínua resulta na qualidade dos elementos do sistema, e por sua vez no produto final, quer seja ele um produto consumível ou um serviço prestado [35, 36].

As auditorias podem ser classificadas consoante a organização, ou seja quem irá realizar a auditoria, e quanto ao âmbito, ou seja sobre a alçada de uma determinada área em específico.

Quanto à organização, as auditorias poderão englobar 3 tipos: Auditoria interna, quando uma organização realiza uma auditoria sobre o seu sistema e os seus respetivos métodos, utilizando então auditores internos e ferramentas próprias; Auditoria externa, que é realizada para apoiar a auditoria interna e validar a melhoria contínua, esta é caracterizada pela realização por parte de uma empresa externa, que por sua vez é imparcial e utiliza ferramentas mais específicas para a verificação; e finalmente, a Auditoria extrínseca, que é realizada pelo órgão máximo da certificação, e tem como objetivo a certificação internacional da organização e a comprovação da conformidade do produto final ao nível internacional.

No que diz respeito às classificações quanto ao âmbito, estas podem ser: referentes ao sistema, que engloba análise de fluxo do sistema e dos processos envolventes; alusivas ao processo, colocando em ênfase a verificação dos procedimentos de execução e o respetivo controlo da qualidade; ou relativo ao produto, que corresponde à verificação da adaptação de fabrico e consumo às especificações e/ou normas [37].

3.2 Ferramentas de Auditoria

Tal como foi visto, uma auditoria deve ser realizada com base em avaliação de factos e outros tipos de informações relevantes. De modo a que se garanta que a avaliação é realizada de forma imparcial e com equidade, criaram-se as denominadas ferramentas de auditoria. Dizem-se ferramentas de auditoria qualquer instrumento que o auditor possa utilizar de forma a atingir os objetivos da auditoria [38]. Ishikawa, impulsionador da gestão da qualidade, em 1984, apresentou numa das suas obras, denominada de "what is total quality control?", as ferramentas existentes relacionadas com métodos estatísticos, e por sua vez subdividiu-as consoante o grau de dificuldade e implementação e utilização [39].

O primeiro grau, inerente à classificação feita por Ishikawa, denominado de método estatístico elementar, demonstra as ferramentas mais simples, e por sua vez mais usuais. A utilização dessas ferramentas por si só, já identificam problemas de processos, ou de estruturação de serviços. Dentro dessa categoria encontram-se ferramentas como diagrama de Pareto, diagrama causa efeito, folhas de verificação (*checklists*), entre outros.

Os métodos estatísticos intermédios, segundo grau de classificação, contemplam

ferramentas de utilização mais específica e não muito habitual, alguns exemplos de ferramentas são a teoria de amostragem e os métodos de estimação de testes. Por fim, encontram-se os métodos estatísticos avançados que são pouco empregues devido à complexidade inerentes aos mesmos. Exemplos dessa categoria são a análise multivariada e o método de investigação operacional [39].

A estatística desempenha um papel bastante importante na gestão da qualidade, oferecendo suporte para realizar distinções e variações entre produtos, processos ou sistemas de gestão. Contudo a estatística não é a única forma de validar/ assegurar a conformidade de um produto ou serviço, outros métodos de validações de dados verbais e/ou raciocínios também existem [40].

Pelo disposto no parágrafo prévio, a vastidão da aplicação das ferramentas em auditoria, vai mais além dos métodos estatísticos. A necessidade de garantir a organização de dados verbais, exposição de raciocínios e formas de planeamento, culminou na formação de um novo grupo de ferramentas destinadas à organização de dados verbais. Nesse grupo são incluídos ferramentas como diagrama de afinidades, diagrama de relações, matrizes de prioridades, entre outros [39].

Não obstante ao referido anteriormente, no início dos anos 90 notou-se uma forte aposta na automatização de processos, principalmente nas grandes empresas, que viam as novas tecnologias como uma solução na melhoria da eficiência. Esse crescimento levou também ao uso de tecnologias de informação nos processos de auditoria. Manson, afirma que esse novo tipo de ferramentas reduz o tempo efetivo de análise e gestão da informação, aumenta a imparcialidade e reduz erros, assegurando assim um processo de auditoria de qualidade constante [41, 42].

A automatização dessas ferramentas de um modo geral traduziu apenas na adequação das antigas ferramentas em *software*, as suas especificidades também foram aumentadas consoante a necessidade, modificando a classificação das ferramentas de auditoria para um outro nível.

Lyra, num dos seus trabalhos, apresentou um método de classificação dessas ferramentas tecnológicas [43]. A classificação proposta tem por base o propósito inicial da ferramenta bem como a sua posterior aplicação.

Consequentemente, foram referidas três categorias. A primeira categoria abrange as ferramentas generalistas de auditoria, ou seja, *software* que consegue simular determinadas situações; analisar documentos; processar amostras; e apontar duplicidades em ambientes generalizados, fornecendo assim uma assistência ao nível do processamento de dados generalistas. Contudo, esse *software* não consegue aplicar funções específicas, como cálculos complexos, e não possui a capacidade de reconhe-

cimento de situações.

Outra categoria referente à classificação realizada por Lyra, denomina-se ferramentas de utilidade geral, que é um tipo de *software*, embora generalista, não foi criado especificamente para auditorias, isto é, é capaz de realizar funções comuns como simples cálculos ou armazenamento de dados, que por sua vez também fazem parte do processo de auditoria. Exemplos de ferramentas que se inserem nessa categoria são o Excel e o MySQL. Por fim, encontra-se a categoria das ferramentas especializadas, *software* particularmente desenvolvido para uma determinada área, reunindo assim todas as condições necessárias para a realização de auditorias bastante específicas sem a necessidade de um perito na área. Uma das principais desvantagens dessa categoria, é que devido à especificidade, o *software* podem ser bastante dispendioso [43, 44].

Singleton, em 2006, refere no seu artigo que o *software* mais utilizado para o processo de auditoria é o que se encontra englobado na categoria de ferramentas generalizadas. Razões como redução do tempo e de custos, ou como redução de tempo de processamento de documentos sem perder o foco da auditoria, suportam tal afirmação [45]. Contudo ferramentas específicas para a auditoria também são desenvolvidas, sendo um desses exemplos o programa "Auditor".

O programa Auditor foi desenvolvido pelo Dr. Fernando Lopes (H.S.João/FMUP) no início da década de 90 e foi distribuído pelo então Sistema de Informação para a Gestão de Serviços de Saúde (SIGSS), posteriormente denominado de Instituto de gestão informática e financeira da saúde (IGIF) e atualmente Administração Central dos Sistemas de Saúde (ACSS), a todos os hospitais do SNS com o objetivo principal de auxiliar nas auditorias internas da codificação clínica. Assim, o programa Auditor é uma ferramenta informática, desenvolvida em FoxPro 2.6 num ambiente MS DOS, destinada a percorrer ficheiros, extraídos da base de dados dos GDH, e verificar a conformidade dos códigos do CID-9-MC, dos dados administrativos, da classificação em GDH, e conseqüentemente as suas relações [40].

O sistema operativo DOS foi o primeiro a ser desenvolvido aquando da explosão do mercado de computadores em 1980. Características como permitir um único utilizador, os comandos somente eram introduzidos pelo teclado e a sua interface não suportar janelas, são grandes limitações deste sistema operativo [46].

A nível de funcionamento o programa Auditor possibilita inicialmente um menu principal que conta com 3 opções, tal como demonstrado na figura 3.1(a). A primeira opção ativa um menu de configurações, 3.1(b), nele é possível alterar diretórios tanto dos ficheiros temporários quanto dos *outputs* finais, permite também a

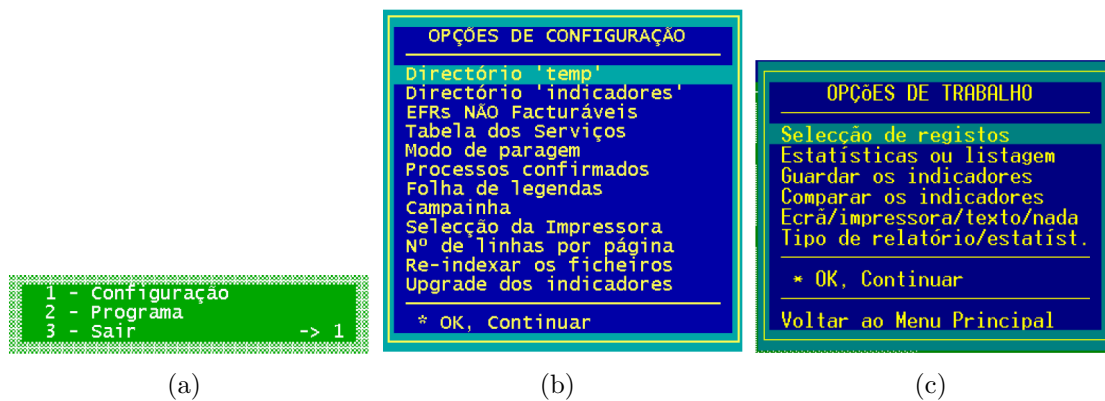


Fig. 3.1: Menus de Trabalho do programa auditor [47]. (a) Menu Principal; (b) Menu de Configuração; (c) Menu de Trabalho.

modificação do modo de funcionamento das abreviaturas utilizadas na codificação, como ainda permite a configuração da impressão do *output*. Essas configurações são utilizadas por forma a adaptar o *software* a cada utilizador. A terceira opção é para sair do programa.

A segunda opção que é a base do programa, permite executar uma auditoria. Para isso é necessário efetuar a seleção de um ficheiro, que por sua vez deverá estar no formato correto. Após essa etapa é então apresentado um menu de trabalho, tal como exposto na 3.1(c), nele é possível seleccionar o modo de execução da auditoria, por exemplo, se a auditoria irá ser realizada em todos os registos, ou somente numa amostra específica. Também é possível optar por um *output* somente com dados estatísticos, ou com uma listagem descritiva. Outra possibilidade é seleccionar que os dados do *output* sejam comparados com outros de uma auditoria diferente, criando assim indicadores [47].

O programa auditor possui algumas limitações, muito devido ao antigo ambiente de desenvolvimento, tais como:

- A plataforma de desenvolvimento de aplicações FoxPro foi descontinuada pela Microsoft, e só será suportada pela mesma até 2015;
- Computadores mais recentes de 64 bits, não permitem o funcionamento de programas em ambiente DOS, a alternativa é a instalação de emuladores DOS;
- A integração do programa com outros recursos é muito limitado, por exemplo, a integração com a impressora;
- Em relação à interface, não é de todo amigável, pois obriga ao uso de diversas

associações realizadas através do teclado, tornando a utilização do programa bastante difícil e levando à necessidade da realização de cursos de iniciação à utilização do programa;

Diversas versões já foram desenvolvidas, sendo a primeira realizada em 1996 e a última versão em Junho de 2013 [40]. Essas atualizações são realizadas de modo a que o programa esteja de acordo com as novas diretrizes e novas versões dos agrupadores de GDH e do sistema de classificação CID.

3.3 Utilização de Sistemas Periciais na área médica

Os Sistemas Periciais (SP) ao longo dos anos foram sendo desenvolvidos em diferentes linguagens. Assim, com o progresso na área das tecnologias verificou-se uma forte aposta em novas linguagens, como por exemplo em linguagem natural, surgindo assim aplicações baseadas em algoritmos e metodologias na área da Inteligência Artificial [38]. A evolução em direção da Inteligência Artificial começou em 1956, essa trajetória histórica possui foco no desenvolvimento de uma linguagem sofisticada e semelhante à forma natural de raciocínio [48]. A área da Inteligência Artificial com mais aplicações na área de auditoria são os Sistemas Periciais [49]. Ainda nesta secção serão apresentadas as principais características dos SP.

A aplicação dos sistemas periciais, era mais focalizada na manipulação de objetos através de "máquinas inteligentes", que por sua vez substituíam profissionais com domínio em ferramentas bastante específicas, contudo este propósito obteve um benefício comercial relativamente baixo devido a baixa flexibilidade dessas máquinas [50]. Durante as últimas três décadas, a utilização dos sistemas periciais tem vindo a emergir com um importante papel na área das ciências da informação. A sua utilização em ferramentas práticas, como na área da auditoria, teve um forte crescimento na comunidade científica [51].

Na literatura, é possível encontrar diversos autores que abordam o conceito de Sistema Pericial. Por sua vez existem vários conceitos divergentes e/ou complementares. Um exemplo é o conceito referido por Jackson, em 1999, que diz que um Sistema Pericial é um sistema informático que utiliza conhecimento para a resolução de problemas. Esse sistema raciocina através do conhecimento fornecido por um ou mais peritos, e tem como finalidade a resolução de um problema ou o aconselhamento dentro de uma área específica [52].

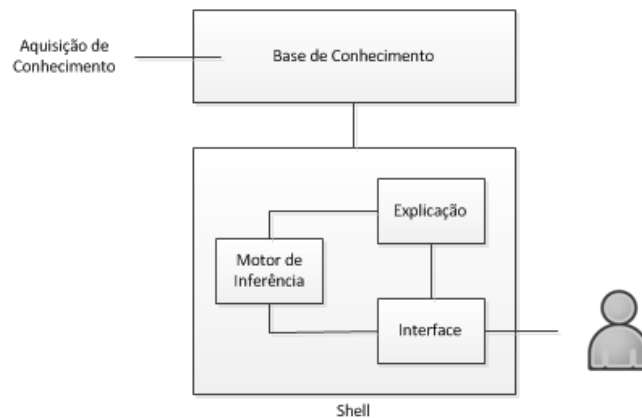


Fig. 3.2: Arquitetura de um sistema Pericial

Entre os vários conceitos existentes na literatura, é possível destacar determinadas características comuns, como: o facto de o Sistema Pericial ser um programa de computador que tem tendência a se comportar como um perito humano; o facto de possuir conhecimento de um ou mais peritos que possibilita por sua vez a resolução de problemas em áreas específicas; o facto de possuir a vantagem de explicar o seu raciocínio tornando-se numa ferramenta mais confiável [53].

Essas características específicas dos sistemas periciais tornam-nos numa ferramenta diferencial, e por sua vez bastante adequada para a implementação na área da auditoria, é possível verificar um exemplo desse aspecto através de Ye, que demonstra no seu artigo o valor da explicação do raciocínio nos sistemas periciais [54]. Ye demonstra que devido à natureza de uma auditoria, é pouco provável que um auditor que é responsável pela auditoria, aceite uma recomendação de um sistema, caso não a perceba. Com a explicação é possível seguir o raciocínio efetuado pelo sistema aproximando assim o auditor à aplicação, e tornando-a mais confiável e por sua vez com melhores resultados [54].

A arquitetura dos Sistemas Periciais, visível na figura 3.2, é normalmente composta por duas grandes componentes: A base de conhecimento e a Shell do sistema [55].

A base de conhecimento é dependente do domínio e a sua correta aplicação e estrutura é vital para o funcionamento eficiente do sistema. Os dados inseridos na base de conhecimento são o resultado do processo de aquisição de conhecimento. O processo de aquisição de conhecimento pode ser efetuado através de estudos de dados, sejam eles numéricos, textos ou vídeos, como também através de entrevistas

com o perito, observação e acompanhamento do perito em ação.

A shell do Sistema Pericial engloba o Motor de Inferência, a explicação ou também denominada de sistema de justificação, e a interface com o utilizador. A shell tem como objetivo processar os pedidos do sistema, traduzir e executar as regras, resolver possíveis conflitos e demonstrar as conclusões ao utilizador [55]. O motor de inferência trabalha em conjunto com a base de conhecimento por forma a desenvolver o pensamento lógico visando a gerar uma conclusão lógica. O sistema de justificação é a mais valia dos SP e tem como função criar justificativas que explicam a conclusão alcançada. Por último a interface que é a camada de contacto com o utilizador, esse contacto pode ser realizado através de perguntas e respostas ou até mesmo com a demonstração da conclusão [56].

Os SP podem assim assumir diversas funções dentro da área da auditoria, porém os designados para o controlo de qualidade têm maior destaque ao nível de investigações práticas devido à necessidade de controlo e automatização dos sistemas, com vista à melhoria de rendimento [57]. Diversos autores investigaram e desenvolveram sistemas nesse âmbito, como é o caso de Moore, que em 1995 desenvolveu um SP voltado para o controlo de processos generalistas [58]; ou como Eyada que em 1990 também desenvolveu um Sistema Pericial para auditar processos tendo em conta o controlo tanto dos produtos, como dos processos [59, 60]. Brinck e Mahalingam também desenvolveram dentro desse âmbito, um SP que era capaz de avaliar a qualidade de uma produção em tempo real, através da deteção e posterior correção de erros aquando do processo [61, 62].

Outro tipo de Sistema Pericial aplicado à área da auditoria, são os sistemas de suporte à decisão. Tal como citado por António Tomás, "Risk Advisor" é um SP desenvolvido por Coopers e Lybrank que auxilia na avaliação do risco de execução de uma determinada auditoria, identificando todos os processos/procedimentos em risco [63]. Outro sistema também referido por António Tomás denomina-se RICE (*Risk Identification and Control Evaluation*), o seu propósito é a identificação dos riscos de uma determinada organização e apresentando-os em indicadores. Esses indicadores poderão, posteriormente, serem utilizados pelo auditor por forma a planear a auditoria.

Wagner, no seu artigo, relata outro tipo de sistemas periciais para a auditoria, os que auxiliam a planificação de uma determinada auditoria. Estes não são tão implementados devido à diversidade de áreas de conhecimento [63]. Contudo, Arthur Young desenvolveu o EXPERTEST, um SP que ajuda a produzir cronogramas com as diversas atividades referentes a auditoria. O sistema é capaz de ajustar em 19

áreas distintas e de produzir relatórios que auxiliam o auditor na gestão da auditoria [64]. Outro exemplo também na área do planeamento de auditoria é o EMYCIN que auxilia o auditor na tomada de decisão, como por exemplo em que área se foca, ou em que processos. A base do EMYCIN tenta simular o raciocínio humano através de regras simples "if then else", o EMYCIN foi desenvolvido em LISP e a sua base foi num modelo para a geração atual de sistemas periciais [63, 65].

As aplicações para auditoria baseadas em Sistemas Periciais na área médica, ao contrário das outras áreas, que se baseiam em estatísticas e métodos probabilísticos, baseiam-se em modelos de relações e fatores externos que são descritos por peritos da área [66]. Essa informação fornecida possui uma validação conflituosa, perguntas como "como é possível saber se um determinado perito possui domínio e experiência na informação providenciada?" são suscitadas na comunidade científica. Muito por este aspecto, poucas são as ferramentas de auditoria desenvolvidas no mercado com implementação exclusivamente da área médica, contudo acredita-se que das poucas existentes praticamente todas trazem algum benefício prático [67].

Dois exemplos bastante atuais de aplicações, desenvolvidas para auditorias clínicas, são o SANDS e o ChronoMedIt. Wright descreve no seu artigo que o apoio à decisão clínica reduz o erro médico e conseqüentemente melhora a qualidade do serviço prestado. Assim a sua aplicação denominada de SANDS (*Servise-oriented Architecture for Clinical Decision Support*) traz benefícios pois possui uma arquitetura diferencial adequando no seu motor de inferência protocolos de auditoria de qualidade fornecidos pelo *National Health Information Network* (NHIN), englobando assim diferentes áreas de implementação dentro da área médica. Nos protocolos da NHIN estão descritos fluxogramas de atendimento, *checklists*, entre outras ferramentas de auditoria. Um aspecto importante do SANDS é que consegue lidar com incertezas que possam estar associadas, através da utilização de regras de Bayes, o que torna a aplicação ainda mais voltada para a realidade clínica. O SANDS foi testado em 6 casos de diferentes áreas, e concluiu-se que as suas vantagens, comparativamente com outras arquiteturas de suporte à decisão clínica, passam por modularidade, expressividade no apoio 'a decisão e interoperabilidade [68].

O outro exemplo é o ChronoMedIt (*Cronological Medical Audit*), esse SP tem como função realizar a auditoria e controlo da execução de procedimentos relacionados com as doenças crónicas, como por exemplo, prescrição de medicamentos, período de acompanhamento, tempo de espera e análises de rotina. Os requisitos para a implementação do projeto foram obtidos através de peritos e de observação do ciclo de funcionamento. Erros na prescrição de medicamentos/exames, lapso de

medicamentos e erros na análise de resultados podem vir a ser reduzidos devido à sua *Framework* possibilitar um registo contínuo e apresentado em forma de gráfico. Outro aspecto deste *software* é a criação de relatórios com gráficos de acompanhamento das medicações e/ou medicamentos. Os resultados obtidos em testes com o ChronoMedIt são notáveis, o que comprova o aumento da melhoria da qualidade do serviço aquando da sua implementação [69].

Desenvolvimento do Sistema Pericial

Neste capítulo será abordado o processo de desenvolvimento do Sistema Pericial. Este processo foi baseado numa *guideline* denominada de ESAQAS, elaborado em 1998 por Bayraktar, aquando do seu estudo para o desenvolvimento de um SP para a área da qualidade [70].

Esta *guideline*, exemplificada na figura 4.1, conta com 8 fases que se inicia com a definição do domínio do problema e termina com a apresentação de um protótipo ao utilizador, de modo a verificar a sua aceitação [70]. A sequência e áreas de cada fase do processo de desenvolvimento baseado no modelo de ESAQAS foi adotada ao longo do capítulo, por forma a descrever o processo de desenvolvimento.

4.1 Definição do domínio do problema

O projeto desenvolvido insere-se particularmente num serviço clínico, o serviço de Obstetrícia, que ao nível de organização das áreas clínicas está em conjunto com a Ginecologia.

A escolha da especialidade prendeu-se com o facto de registos associados a obstetrícia serem, aquando da implementação da codificação, os que possuíam mais erros. Estes erros foram maioritariamente ocasionados devido à complexidade da codificação, que por exemplo, levava a códigos associados a Recém-Nascidos (R/N) estarem codificados em registos de maternidade, e vice-versa. Este facto despertou potencial interesse devido à possibilidade da inserção na aplicação de incertezas associadas aos registos clínicos. Porém, atualmente estes erros já não são verificados, pois a prática contínua da codificação pelos médicos codificadores e auditores, possibilita identificar imediatamente se a natureza do registo é de R/N ou de maternidade.

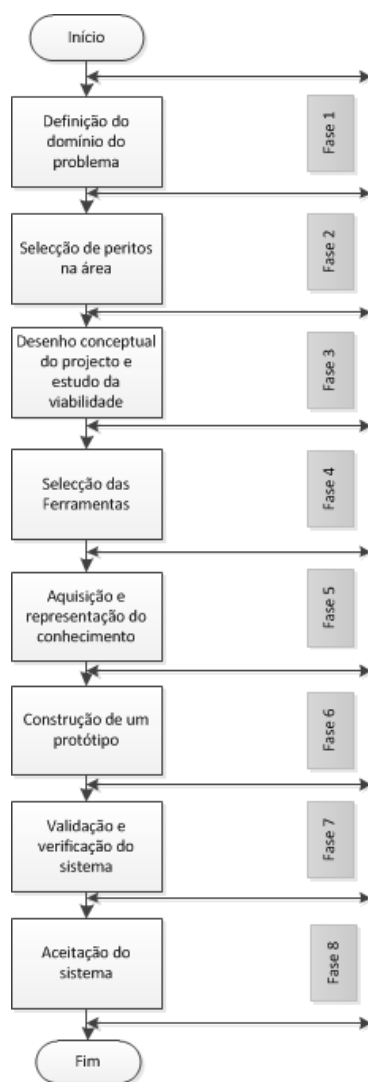


Fig. 4.1: Modelo de desenvolvimento de ESAQAS. Adaptado de [67].

Outro aspecto que levou à escolha da especialidade foram os documentos de apoio, que no caso da obstetrícia estavam completos e representavam aproximadamente 31% de toda a documentação descrita.

De forma a dimensionar o volume de trabalho, diariamente num Hospital central, são realizados uma média de 8 partos. Nesse mesmo hospital, no ano de 2011, foram efetuadas 16251 consultas da especialidade de ginecologia e obstetrícia e nessa mesma área, foram realizados 32564 meios complementares de diagnósticos [71]. Todas essas consultas e meios complementares de diagnósticos, bem como procedimentos cirúrgicos são diariamente codificados no Gabinete de Codificação (GC), e são evidentemente objeto de faturação.

Tal como já foi referido na presente dissertação, por forma a garantir a correta

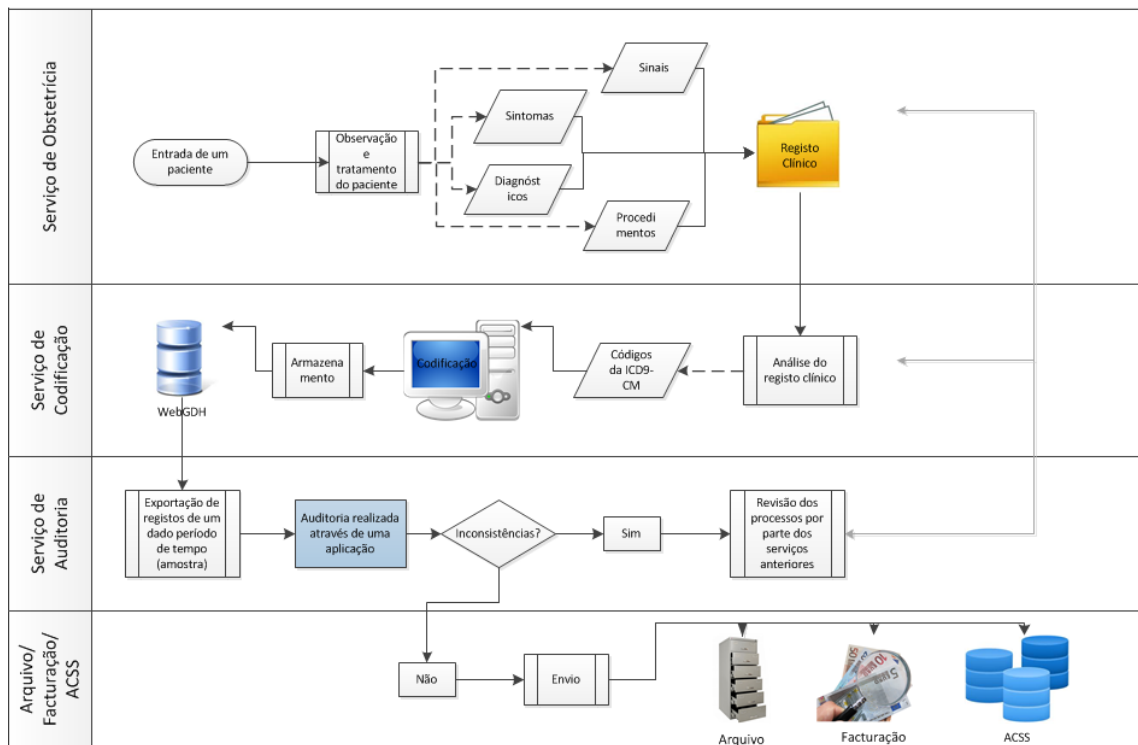


Fig. 4.2: Fluxo do registo clínico no Hospital.

faturação, auditorias internas são frequentemente realizadas. Para isso, foi criado um fluxo de trabalho que garante a qualidade da codificação do registo clínico. Esse fluxo exemplificado na figura 4.2 também representa o ciclo de vida do registo clínico.

O ciclo de vida inicia quando um paciente dá entrada num hospital. É então criado um novo episódio hospitalar, e nele é registado todas as observações e consequentes tratamentos efetuado pelo médico. Aquando da alta do paciente, o episódio hospitalar é anexado ao registo clínico, e este enviado para o Gabinete de Codificação. No GC é realizado a tradução dos dados, ou seja a passagem de texto para códigos, através da utilização da CID-9-MC. Esses dados codificados são armazenados na Base de dados clínico-administrativa do hospital.

Quando é realizado uma auditoria, é feito uma exportação dos dados codificados. Caso a auditoria demonstre inconsistências é então realizada uma revisão de todos os processos dos serviços suprajacentes. Caso não possua inconsistências, os dados são enviados para faturação, exportados para a base de dados central da ACSS para posterior construção de dados epidemiológicos, e enviados para o arquivo do hospital.

4.2 Selecção do perito

À utilização da tecnologia de informação como forma de auxílio ao perito, deve-se ao facto da dificuldade deste último, analisar em tempo útil as inúmeras combinações de não conformidades de registos clínicos. A morosidade pericial pode ser minimizada se a informação e modo de raciocínio estiverem dispostos numa base de conhecimento, auxiliando assim o perito no processo de auditoria [72].

Por conseguinte, é necessário então uma correta seleção do perito, e por sua vez a validação do seu conhecimento, por forma a que o sistema possua um desempenho uniforme [70, 73]. Assim, e por forma a implementar o referido anteriormente, esta fase foi dividida em duas etapas: a escolha de um perito humano, e a escolha de fontes bibliográficas que suportassem os conhecimentos do perito.

Na seleção dos recursos escritos foram seleccionados artigos que descrevessem técnicas de desenvolvimento de sistemas periciais para a garantia da qualidade [57, 58, 59, 61, 70], bem como bibliografia que suportasse o conhecimento na área da codificação [11].

A seleção do recurso humano, o perito envolvido nesse projeto, muito devido ao seu conhecimento na área, foi o Dr. Fernando Lopes, licenciado na área da Medicina pela FMUP em 1981, funcionário da Faculdade desde 1985, desempenhando diversas funções desde apoio informático, ensino, ou na investigação científica. Também funcionário do Hospital São João, desde 1988, onde desempenha funções de codificador clínico, auditor interno e responsável pela formação de médicos codificadores.

Além disso, também foi responsável pela criação em 1999 do "InterSim", protótipo de uma base de conhecimento em sistemas de informação em medicina na internet, a par da criação do "Programa Auditor", já referido anteriormente, programa este que atualmente ainda se encontra implementado na maioria dos hospitais do SNS. Acumula ainda funções como consultor na área da codificação na ACSS e da ARS Norte (Administração Regional de Saúde do Norte); e sócio fundador, bem como atual presidente, da Associação dos Médicos Auditores e Codificadores Clínicos (AMACC).

4.3 Desenho Conceptual e estudo da viabilidade técnica

No desenho conceptual questões como "Porquê?", "Para quem?" e "Como?" são definidas. Assim, e em resposta à primeira questão, é-nos perguntado o objetivo da aplicação, e tal como já foi referenciado nessa dissertação, o objetivo é criar uma ferramenta prática através da utilização de um Sistema Pericial, que auxilie o utilizador no processo de auditoria interna dos registos clínicos. Tendo em consideração a afirmação referida anteriormente, a aplicação deverá ser capaz de "pensar" e "agir" como um médico auditor.

Em resposta à segunda questão, os utilizadores seriam os médicos auditores, que além de possuírem conhecimento na área, são os que, devido à elevada especificidade do sistema, podem deduzir e pôr em causa o resultado da aplicação. A nível de território, todos os hospitais do SNS podem receber essa aplicação, dado que esta é adaptada à base de dados dos GDH que por sua vez é idêntica em todo o território nacional, e possui apenas algumas adaptações a nível internacional.

Por forma a responder como seria desenvolvida a aplicação é necessário explicar a arquitetura do sistema a desenvolver, evidenciado na figura 4.3. Primeiramente é muito importante realçar que a aplicação teve como base a ferramenta atualmente em uso - o programa Auditor. Assim sendo, a aplicação deverá ter o conhecimento interno do programa Auditor, contudo traduzida numa linguagem mais prática de traduzir o raciocínio do perito, no caso em particular na programação lógica - PROLOG (PROgrammation en LOGique).

Em suma, será desenvolvido um sistema pericial que possui regras já descritas no programa Auditor e um motor de inferência com encadeamento direto. Esse método de inferência é orientado aos factos, ou seja, analisa um conjunto de factos que estão incutidos num registo clínico e utiliza as regras para deduzir erros de codificação. A utilização deste método prende-se com a similaridade ao raciocínio humano e consequentemente com o raciocínio utilizado pelo perito.

O compilador escolhido para o desenvolvimento da aplicação foi o LPA Prolog da *Logic Programming Associates*, de versão 4.8 e licença registada em nome do GECAD (Grupo de investigação em Engenharia de Conhecimentos e Apoio à Decisão). O LPA Prolog 4.8 inclui *toolkits* que serão utilizadas posteriormente no desenvolvimento, como por exemplo:

- FLEX - O flex é uma *toolkit* para sistemas periciais que comparativamente com

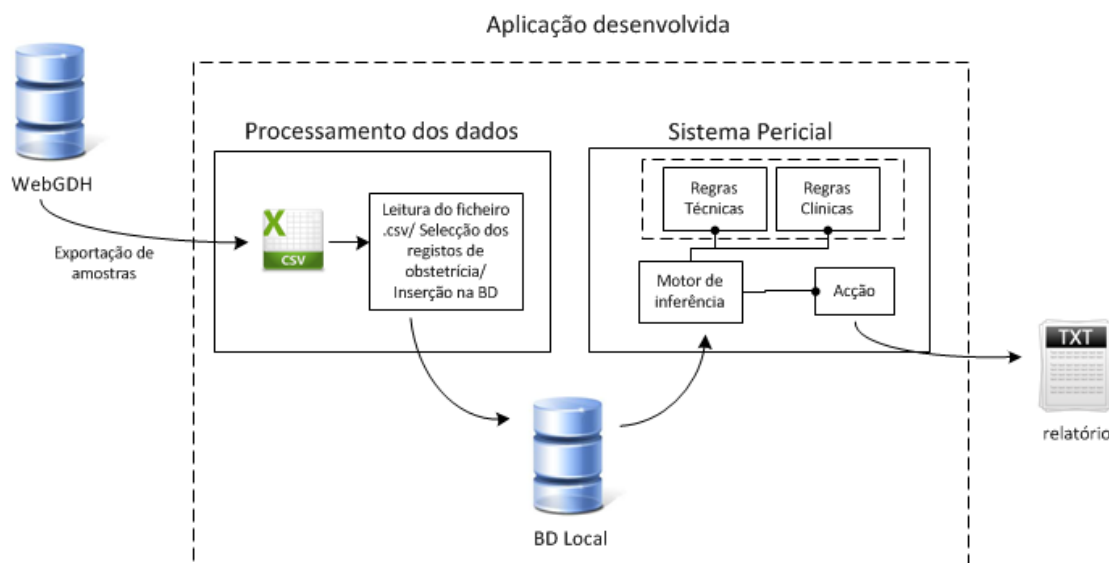


Fig. 4.3: Arquitetura da aplicação desenvolvida.

outras, é bastante versátil e simples. A sua extensão é em KSL (Knowledge Specification language). O Flex é baseado em PROLOG e permite desenvolver regras de alto nível, contudo utiliza uma linguagem natural para definir o conhecimento, o que o torna simples. Por sua vez, o flex possui um mecanismo iterativo de pergunta e resposta, que pode se estender a uma pequena interface gráfica [74].

- PRODATA - O prodata suporta uma ODBC (Open DataBase Connectivity), assim é permitido aceder a partir do PROLOG às tabelas da Base de dados sem qualquer necessidade de adaptações adicionais [75]. Assim ações simples como conexão, inserção e/ou eliminação de dados, tornam-se mais simples.

Em conjunto com a aplicação em PROLOG foi desenvolvido outra pequena aplicação em PHP (PHP: hypertext preprocessor) com o objetivo de processamento dos dados iniciais. Esta aplicação é também apresentada na arquitetura do sistema e será posteriormente detalhada aquando do subcapítulo da construção do protótipo. A nível de base de dados local, foi utilizado o *MySQL* que é um DBMS (*DataBase Management Systems* ou Sistema de Gestão de Base de Dados).

4.4 Selecção de Hardware e Software

Tal como já foi referido anteriormente, a nível de *software*, é necessário possuir uma licença, no caso, para o LPA Prolog. O sistema operativo mínimo exigido pelo

mesmo é o *Microsoft Windows 98*, contudo é recomendado a utilização do *Microsoft Windows 7*. A utilização de um *browser* também se deverá ter em conta.

É necessário também, um servidor independente de plataforma, por exemplo o XAMPP, com base de dados em MySQL, servidor apache e um interpretador de linguagem de *script* em PHP [76].

A nível de *hardware*, os requisitos mínimos recomendados passam por um processador *single-core* a 1GHz, 512 MB de memória RAM e cerca de 20 GB de espaço em disco, que suportará o sistema operativo mais a aplicação e armazenamento de ficheiros.

4.5 Aquisição do conhecimento e representação do Conhecimento

Os Sistemas Periciais são distintos de outras áreas da Inteligência Artificial pois são constituídos com o objetivo de resolver problemas de uma forma bastante similar aos peritos humanos [77].

A aquisição do conhecimento é um processo que normalmente está incutido no desenvolvimento de todas as aplicações baseadas no conhecimento de outrem. As técnicas de aquisição de conhecimento são variadas, desde estudo de vídeos e textos, aprendizagem através de dados ou de entrevistas com os peritos humanos [78].

No caso em particular, a técnica de aquisição de conhecimentos utilizada foi a entrevista com o perito. Nelas foi possível descobrir a origem dos ficheiros de amostra do sistema, que por sua vez eram geradas a partir de uma exportação de uma base de um hospital central. Esta amostra é fornecida em formato CSV (*Comma separated values*), ou seja, valores separados por vírgulas, e possui um conjunto de três ficheiros em que os seus nomes são criados automaticamente a partir das condições específicas delineadas no ato da exportação. Os três ficheiros são do tipo: "C", caso o ficheiro contenha códigos da CID-9-CM; "D", caso o ficheiro contenha dados administrativos; ou "S" se o ficheiro possuir dados relacionados com serviços e transferências. Uma particularidade desses ficheiros é a irregularidade de chaves primárias ou identificações principais, ou seja, no ficheiro do tipo "D" uma linha corresponde a um registo clínico, enquanto por exemplo no ficheiro do tipo "C" um registo possui várias linhas, pois são os códigos dos diagnósticos e/ou procedimentos efetuados num episódio do doente. O formato do ficheiro viria a trazer alguns problemas relacionados com a seleção de determinados campos, bem

como iria tornar a análise mais lenta, comparativamente com uma análise a campos específicos numa base de dados MySQL.

Também através das entrevistas foi possível realizar um levantamento de todas as regras a validar na área de atuação, e o respetivo raciocínio associado. Posteriormente é então realizada uma validação dos conhecimentos adquiridos através do perito com o auxílio dos recursos escritos, em particular a utilização da CID-9-MC bem como o portal da codificação e dos GDH.

O levantamento de dados e posterior análise totalizou 168 regras, dessas 132 de carácter técnico e 36 de carácter clínico. Esta documentação está disponibilizada no anexo A.

A representação do conhecimento é um subprocesso da aquisição do conhecimento [78]. No caso a representação escolhida foi do tipo *if-then-else*. Outros tipos de representações também poderiam ser utilizadas, como algoritmos matemáticos, ou lógicas booleanas, contudo a representação *if-then-else* é o método mais aproximado do raciocínio humano [70]. Como a aplicação é enquadrada no âmbito da auditoria, particularmente as conclusões das regras criadas serão avisos de não conformidades, ou seja, alertas que serão demonstrados num *output* juntamente com a identificação do registo não conforme, de modo a que o auditor consiga executar a verificação e possível alteração.

O raciocínio transcrito após a aquisição do conhecimento e da validação do mesmo, representa-se através de um conjunto de perguntas e respostas, cujo objetivo é detetar a não conformidade do registo num determinado campo. Esse raciocínio é demonstrado na figura 4.4.

Com o levantamento de dados e a transcrição de todas as regras foi possível esclarecer alguns pormenores de execução do protótipo, como por exemplo: a demonstração do *output*, como se iria criar ciclos de funcionamento, ou até mesmo como agrupar a leitura por registos.

4.6 Construção de um protótipo

A construção de sistemas segue normalmente modelos de construção. A Engenharia de *Software* nasceu da necessidade dessa implementação, e possui como objetivo aplicar uma abordagem sistemática, disciplinada e quantificável, para o desenvolvimento operação e/ou manutenção de um *software* [79]. Os modelos criados a partir de técnicas de análise da Engenharia de *Software* são inúmeros, contudo é possível dividi-los em duas grandes metodologias: as tradicionais e as ágeis.

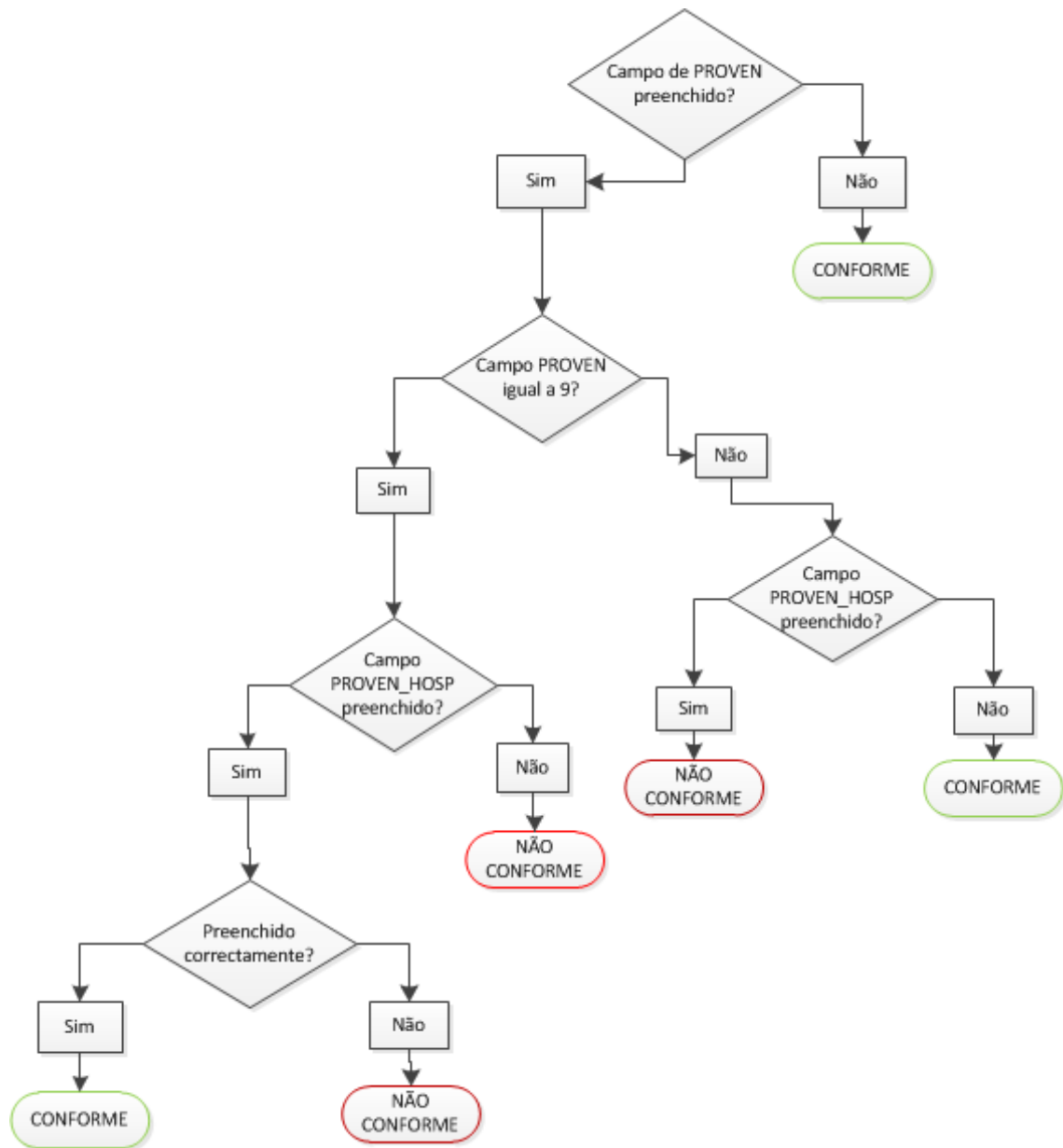


Fig. 4.4: Transcrição do raciocínio da validação técnica da proveniência do paciente de outro hospital.

A construção do protótipo baseou-se em metodologias ágeis, pois estas possuem características como o desenvolvimento iterativo e incremental, documentação extensiva, e enfoque no produto final, tornando assim maiores as probabilidades do sistema atender aos requisitos necessários, pois é um método mais flexível comparativamente às metodologias tradicionais [80].

4.6.1 Construção da Base de Dados Local

A construção da base de dados local foi baseada nos ficheiros CSV, em que cada tabela corresponde a um ficheiro e os campos são consoante a disposição entre vírgulas dos dados.

Cada ficheiro traduz-se em duas tabelas: uma tabela denominada auxiliar, que contém todos os registos exportados da BD do Hospital num determinado período de tempo, e outra tabela denominada de principal que contém somente os registos que fazem parte da obstetrícia, ou seja, registos de mães e R/N. O modelo de dados é visível na figura 4.5. É importante realçar que as tabelas auxiliares surgiram devido à especificidade da análise das regras, pois como se trata de uma área bastante específica, e por forma a otimizar a aplicação, fez-se a tabela principal somente com os registos a analisar, ao passo que a auxiliar com todos os registos. A escolha da utilização de uma nova tabela ao invés da eliminação dos dados, prende-se com o facto de reduzir o número de erros [81].

No caso dos dados administrativos o número de episódio é único, ou seja, um registo possui somente um número de episódio. Todavia na tabela de codificação e serviços um registo contém várias linhas de códigos associados, que por sua vez irão repetir em cada uma os números de episódio, ou seja, por mais que seja único, não se reflete somente numa linha. Assim estas tabelas também possuem como chaves um ID (número *auto-increment*) que é único e diferenciado por linha.

4.6.2 Construção de uma aplicação para o tratamento de dados

Esta aplicação surgiu quando, em sequência da aquisição de conhecimento, se verificou que o ficheiro de entrada na aplicação (ficheiro de exportação da base de dados do hospital) era em formato CSV. Este formato dificultaria a conexão com o LPA

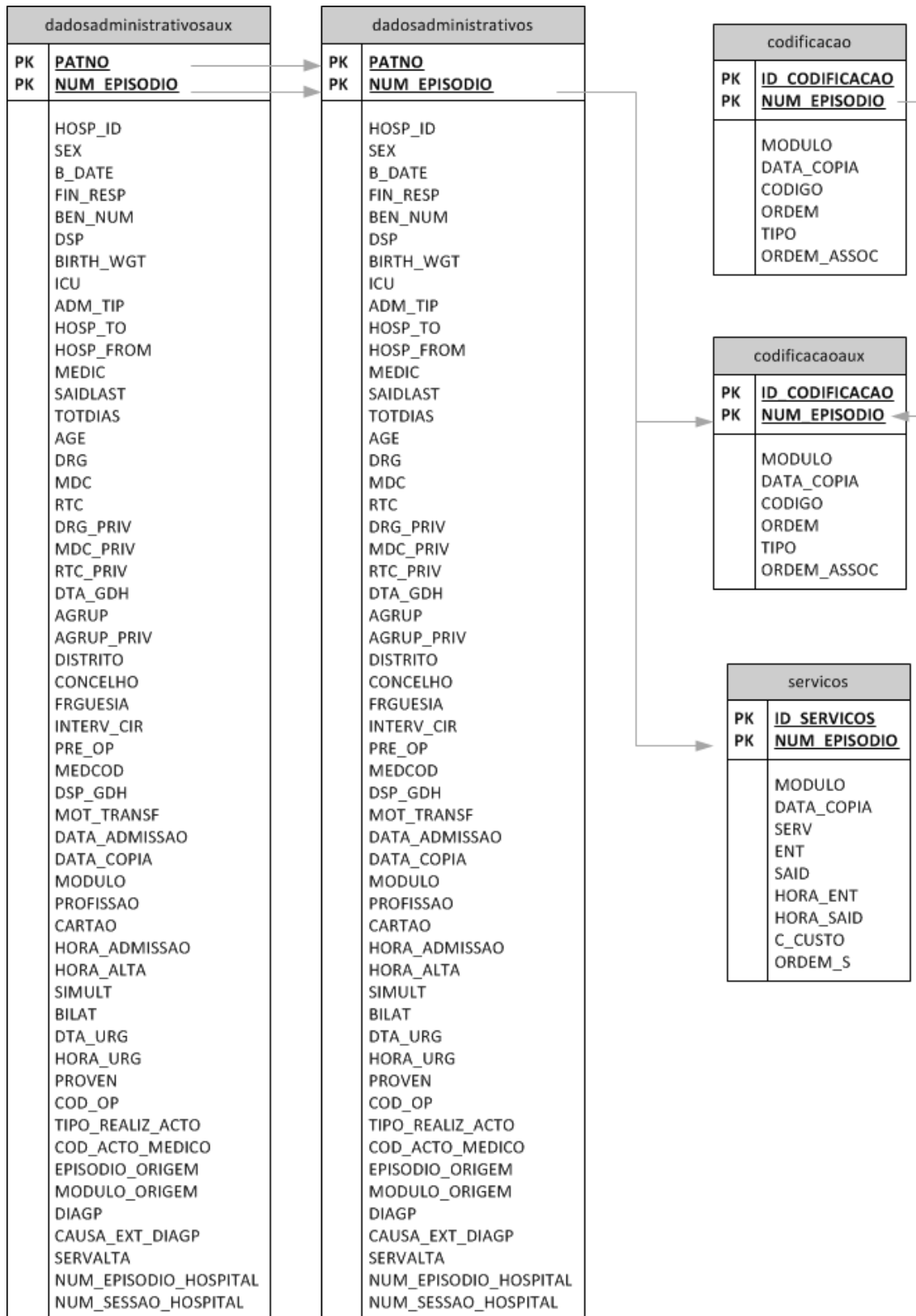


Fig. 4.5: Modelo de dados do Sistema Pericial baseado nos ficheiros de exportação

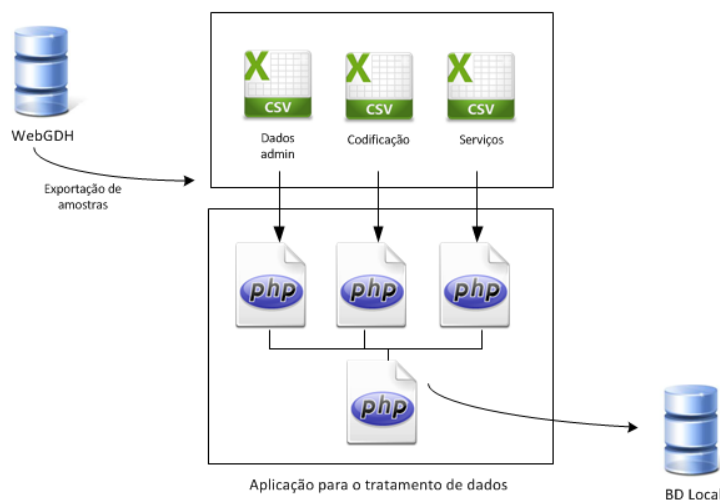


Fig. 4.6: Arquitectura da aplicação para tratamento de dados.

PROLOG, que por sua vez, tornaria a análise mais pesada. Pois a par das *queries* SQL, o LPA PROLOG, ou mais propriamente o PRODATA, não suporta outro tipo de *queries*, no caso em particular, com funções para CSV [75].

A aplicação foi desenvolvida em PHP e tinha como objectivo principal passar os dados do ficheiro CSV para a Base de dados local em MySQL. A sua estrutura é constituída por três ficheiros principais que fazem a ligação aos respetivos três ficheiros criados em formato CSV aquando da exportação. Essa arquitetura é possível verificar na figura 4.6.

Em específico nestes ficheiros, é primeiramente utilizado uma função para abrir o ficheiro da amostra em modo de leitura. De seguida é utilizada a função "getcsv" por forma a ler cada linha do ficheiro e armazenar os valores separados por vírgulas. O modo de leitura é também definido, ou seja, será executado de baixo para cima no ficheiro (registo mais recente para o mais antigo) e será ainda eliminado a última linha, que por sua vez contém o cabeçalho que não é relevante para o armazenamento na BD. Código exemplificado na figura 4.7.

Por forma a armazenar na base de dados é realizada uma *query*, demonstrada na figura 4.8(a) que faz o "insert" linha a linha de cada valor separado por vírgula no ficheiro CSV. Essa primeira *query* armazena todos os valores na BD local numa tabela denominada auxiliar. Posteriormente são realizadas outras *queries* mais específicas. A primeira, figura 4.8(b) insere somente os registos considerados de maternidade, a outra insere somente os registos considerados de R/N. Esses registos são inseridos numa tabela principal.

```

$abra = fopen("TDSR291C.csv","r"); // abrir o ficheiro como modo de leitura

if(!$abra){
    echo("<p> Arquivo não encontrado </p>");
}else{
    while($valores = fgetcsv($abra,10000,",")){
        $dados_file[]=$valores;
    }
}

if(sizeof($dados_file)>1){ // tem de conter conteúdo o ficheiro
    $dados_file=array_reverse($dados_file);

    array_pop($dados_file);
    foreach($dados_file as $values){ // construção de uma maneira para interagir entre o array

        $DB = new ConexaoBD(); // conexão com a base de dados
        $DB->ligarBD($NOME_BD, $USER_DB, $PASS_DB, $NOME_SERVIDOR);
    }
}

```

Fig. 4.7: Código com função específica para leitura do ficheiro CSV

É necessário que se realize o armazenamento ou importação dos dados relativo ao ficheiro administrativo, visto que é a única tabela em que uma linha corresponde a um registo clínico. As *queries* específicas de inserção dos dados de codificação e de serviços (restantes dois ficheiros), figura 4.8(c) utilizam os IDS na tabela de dados administrativos e inserem todas as linhas da tabela auxiliar que possuam IDS semelhantes.

Assim, é possível o preenchimento tanto das tabelas auxiliares com as primeiras *queries* associadas a leitura da função CSV, como nas tabelas principais, que se realizam ou através da análise de certos campos específicos, ou através de IDS (número de episódio) associados.

4.6.3 Construção da conexão à Base de Dados Local

A construção da conexão à Base de dados foi realizada em PROLOG, com o auxílio do PRODATA (*toolkit* do LPA PROLOG). Para carregar o PRODATA é necessário correr um ficheiro DLL (dynamic-link library ou biblioteca de vínculo dinâmico) específico do sistema LPA PROLOG, para isso é imprescindível executar um comando, descrito logo no início do código: "ensure_loaded(system(dblink))." [75].

O comando para inicializar a ligação à base de dados, denominado de "init_db_conn" realiza uma ligação ao *datasource* identificado por "spauditoria", esta ação é feita através do predicado *db_connect/2*. Também neste comando, é realizado as associações, através do "db_attach", dos predicados criados com as respetivas tabelas principais. Por fim são ainda modificadas as definições, por forma a não mostrar as *queries* SQL criadas aquando da ligação.

```
// construção da query
...
$result = "INSERT INTO `spauditoria`.`dadosadministrativosaux` (`HOSP_ID`,`PATNO`,
`SEX`,`B_DATE`,`FIN_RESP`,`BEN_NUM`,`DSP`,`BIRTH_WGT`,`ICU`,`ADM_TIP`,`HOSP_TO`,`HOSP_FROM`,
`MEDIC`,`SAIDLAST`,`TOTDIAS`,`AGE`,`DRG`,`MDC`,`RTC`,`DRG_PRIV`,`MDC_PRIV`,`RTC_PRIV`,`DTA_GDH`,`AGRUP`,`AGRU
VALUES ('".$values[0]."',"$values[1]."',"$values[2]."',"$values[3]."',"$values[4]."',
"$values[5]."',"$values[6]."',"$values[7]."',"$values[8]."',"$values[9]."',"$values[10]."',
"$values[11]."',"$values[12]."',"$values[13]."',"$values[14]."',"$values[15]."',"$values[16]."',
"$values[17]."',"$values[18]."',"$values[19]."',"$values[20]."',"$values[21]."',"$values[22]."',
$db->insertSQL($result); // função de inserção na base de dados
```

(a)

```
// construção de uma query para selecionar e inserir noutra tabela somente os registos de obstetrícia
...
$seleciona = "INSERT INTO `spauditoria`.`dadosadministrativos` (`HOSP_ID`,`PATNO`,`SEX`,`B_DATE`,`FIN_RESP`,
`BEN_NUM`,`DSP`,`BIRTH_WGT`,`ICU`,`ADM_TIP`,`HOSP_TO`,`HOSP_FROM`,`MEDIC`,`SAIDLAST`,`TOTDIAS`,`AGE`,`DRG`,
SELECT * FROM `dadosadministrativosaux` WHERE ((`DIAGP` BETWEEN '630XX' AND '679XX') OR (`DIAGP` = 'V222')
OR (`DIAGP` = 'V240'))";
$db->insertSQL($seleciona);
```

(b)

```
$insere= "INSERT INTO `spauditoria`.`codificacao` (`ID_CODIFICACAO`, `NUM_EPISODIO`, `MODULO`, `DATA_COPIA`,
SELECT `codificacaoaux`.`ID_CODIFICACAO`,`codificacaoaux`.`NUM_EPISODIO`,`codificacaoaux`.`MODULO`,`codifica
FROM `codificacaoaux` INNER JOIN `dadosadministrativos` ON `codificacaoaux`.`NUM_EPISODIO`=
`dadosadministrativos`.`NUM_EPISODIO`;
$db->insertSQL($insere);
```

(c)

Fig. 4.8: Queries de inserção na Base de dados. (a) Query de inserção na tabela auxiliar; (b) Query de inserção dos dados específicos de obstetrícia na tabela principal; (c) Query de inserção específica dos dados de codificação, com o auxílio dos números de episódio.

```
init_bd_conn:-
  db_connect( spauditoria, HDBC ), % efectua ligação ao datasource identificado por 'SPAuditoria'
  assert( datasource( HDBC ) ),
  db_attach( dadosadmin, 'dadosadministrativos'),
  db_attach(codifica,'codificacao'), % associa o predicado codifica à tabela codificacao
  db_attach(servico, 'servicos'), % associa o predicado servico à tabela servicos
  db_flag( show_sql,_,off ). % não mostra as queries SQL geradas

end_bd_conn:-
  retract( datasource( HDBC ) ),
  db_disconnect( HDBC ).
```

Fig. 4.9: Extrato de código PROLOG com comandos para inicializar e encerrar a ligação a base de dados

Etiquetas

Um desses procedimentos auxiliares foi a inserção de etiquetas. Essas etiquetas têm como objetivo uma identificação do registo no programa, por forma a ser possível a realização de ciclos.

Os ciclos são normalmente descritos com uma função bastante específica denominada de "ruleset". No mesmo, é possível invocar um motor de inferência que contém um conjunto de regras definidas, é possível também definir condições de finalização das regras, como escolher o método de seleção das regras ou o método de atualização da agenda entre outras condições.

Os ciclos são definidos pelo método de atualização de agenda, estas atualizações são através da remoção ou promoção das regras disparadas. No caso da aplicação, como as regras precisam de ser disparadas mais do que uma vez para a validação de um conjunto de registos clínicos, o método de atualização de agenda por si só, não é suficiente.

Surgiu então a necessidade de criar etiquetas associadas a cada identificação do registo clínico. Para isso é realizado inicialmente a seleção de todos os números de episódio existentes (identificação dos registos) e o armazenamento desses números numa lista. Posteriormente essa mesma lista irá conter sublistas com as identificações das regras que já foram validadas.

No início de cada regra é feita uma validação se a identificação consta ou não, eliminando assim a repetição de registos clínicos numa mesma regra. Caso a regra seja disparada, no final é realizado uma atualização com a identificação da regra, no campo que contém a identificação do registo (número de episódio). O código associado a este método está descrito na imagem 4.11, e é executado em PROLOG.

Assim, com o método das etiquetas associado ao método de atualização de agenda descrito na *ruleset*, é possível criar ciclos em que todas as regras apropriadas são validadas em cada registo clínico.

Output

As conclusões das regras são alertas de não conformidade de um determinado registo. Contudo a forma com que estes alertas são demonstrados é fundamental. Assim o *output* foi baseado no relatório do programa Auditor, devido aos utilizadores já estarem habituados ao método de demonstração de resultados. O *output* é a impressão de um alerta a demonstrar o tipo de erro, semelhante ao programa Auditor, a identificação do registo não conforme (por forma a facilitar a consulta) e uma explicação

```

##### Predicados associados a atribuição de etiquetas

%% Preenchimento automatico do ID da tabela com os numeros de episódio

procuranumepi:-
    findall(NUMEPISODIO,getnumepisodio(NUMEPISODIO,IDPAT),L),
    insere_lista(L).

insere_lista([]).
insere_lista([X|T]):-
    assert(validatecnico(X,[])),
    insere_lista(T).

%% Preenchimento do campo de validação

actualiza(NUMEDI,NOME):-
    validatecnico(NUMEDI,CAMPOVALIDA),
    append(CAMPOVALIDA,[NOME],LC),
    retractall(validatecnico(NUMEDI,_)),
    assert(validatecnico(NUMEDI,LC)).

%% Eliminação das listas

apaganumepi:-
    retractall(validatecnico(_,_)).

```

Fig. 4.11: Predicados de associação e preenchimento automático de etiquetas

da não conformidade (característica principal dos sistemas periciais). Este último é uma das principais vantagens comparativamente ao Programa Auditor.

Para tal, foi necessário a criação de dois predicados de escrita, um referente às regras técnicas e outro às regras clínicas. Em cada predicado é realizado a escrita de argumentos associados ao mesmo, que no caso são, as identificações dos registos, o aviso e a explicação.

Regras Técnicas

A construção das regras técnicas obrigou a criação de um *ruleset* específico. Assim o conjunto das regras técnicas era chamado na ação de um só vez e por sua vez definido pelo prefixo "admin" mais a identificação da regra.

A título exemplificativo a regra "admin_3" é demonstrada na figura 4.12(b). A regra possui um conjunto de predicados e validações auxiliares tais como:

- Predicado "validatecnico" que utiliza uma lista com a identificação do registo a validar;
- Predicado "getsex" vai utiliza como parâmetro principal o campo "SEX" na base de dados, e especificamente com a identificação do registo a validar;
- Predicado "escreve" que realiza o *output* caso a regra dispare;

- Predicado "actualiza" que faz a atualização da regra na sublista da identificação do registo clínico;
- Validação se a regra foi ou não validada naquele registo clínico através de "and 3 is not included in NOME";

A validação técnica da regra "admin_3" realizada no campo "SEX" verifica se o campo está preenchido corretamente, ou seja, se possui 1, 2 ou 3; referente ao sexo masculino, feminino e intersexo, respetivamente. Em FLEX, traduz-se apenas em verificar se o campo está abaixo de 1 ou acima de 3, caso uma das afirmações se verifique, é então realizado um alerta de não conformidade no respetivo registo clínico.

Outras regras precisam de validações mais complexas como por exemplo a validação do campo "ID_HOSPITAL", que para comprovar o correto preenchimento, tem de ser comparado com um conjunto de identificações. Estas identificações, no caso do programa Auditor constam em tabelas de verificação na BD. No caso do sistema Pericial constam num grupo, esses grupos são definidos no documento KSL e possuem comandos de verificação específicos em FLEX. Um exemplo é o demonstrado na figura 4.12a, em que o grupo é denominado de "idhospital" e é feita a verificação através do comando "instance of".

Algumas regras técnicas ainda precisaram de procedimentos auxiliares realizados em PROLOG. Um exemplo foi a verificação de campos com o formato data, demonstrado na figura 4.13. O FLEX não reconhece o formato, e no LPA PROLOG não existe nenhuma função que associe a data. Assim, foi preciso criar um predicado que seleciona o campo data e divide em dia, mês e ano, por forma a realizar as validações. A base incutida no predicado é passar de uma *string* (o que traz da BD) para uma lista, e nessa lista ler carácter a carácter de forma a seleccionar somente o necessário e gravar novamente como uma *string*.

Esta limitação trouxe problemas posteriores com as validações entre datas, contudo serão posteriormente abordados.

Regras Clínicas

As regras clínicas são muito semelhantes às regras técnicas. A par dos procedimentos auxiliares, da criação da *ruleset* e de grupos específicos, é também necessário, especificamente para as regras clínicas, agrupar os registos clínicos. Tal como já foi referido, na tabela de codificação um registo clínico contém várias linhas com

```

rule admin_1_2
  if validatecnico(NUMEPISODIO,NOME) and gethospid(HOSPID,IDPAT,NUMEPI)
  and NUMEPISODIO is equal to NUMEPI
  and 1.2 is not included in NOME
  and HOSPID is not equal to ''
  and HOSPID is not equal to some instance of idhospital
  then escreve('Código de hospital inválido. N.Paciente:',IDPAT,
  'N.Episodio:',NUMEPI,'Valor diferente do possível apresentado na tabela de identificação
  dos hospitais')
  and actualiza(NUMEPISODIO,1.2)
  because('Valor diferente do possível apresentado na tabela de identificação dos hospitais').

```

(a)

```

rule admin_3
  if validatecnico(NUMEPISODIO,NOME) and getsex(SEX,IDPAT,NUMEPI)
  and NUMEPISODIO is equal to NUMEPI
  and 3 is not included in NOME
  and SEX is above 3
  or validatecnico(NUMEPISODIO,NOME) and getsex(SEX,IDPAT,NUMEPI)
  and NUMEPISODIO is equal to NUMEPI
  and 3 is not included in NOME
  and SEX is below 1
  then escreve('Campo sexo preenchido incorrectamente. N.Paciente:',IDPAT,
  'N.Episodio:',NUMEPI,'O valor do campo sexo é de preenchimento obrigatório e deve estar
  compreendido entre 1 e 3, inclusive')
  and actualiza(NUMEPISODIO,3)
  because('O valor do campo sexo é de preenchimento obrigatório e deve estar compreendido entre
  1 e 3, inclusive').

```

(b)

Fig. 4.12: Extracto de código FLEX com regras técnicas. (a) Regra técnica de validação do campo ID_HOSPITAL; (b) Regra técnica de validação do campo SEX.

```

##### Predicados de alteração do formato data -> String to List then String again

get_day(Date,D):-
  name(Date,L), separa_d(L,D).

separa_d([D1,D2,_,_,_,_,_,_,_],D):-
  name(D,[D1,D2]).

#####

rule admin_4_2
  if validatecnico(NUMEPISODIO,NOME) and getbdate(BDATE,IDPAT,NUMEPI)
  and NUMEPISODIO is equal to NUMEPI
  and 4.2 is not included in NOME
  and BDATE is different from ''
  and get_day(BDATE,X) and X is above 31
  then escreve('Campo Data de Nascimento está num formato incorrecto. Nº Paciente:
  ',IDPAT,'N.Episodio:',NUMEPI,'O valor do campo data de nascimento
  deve estar no formato data')
  and actualiza(NUMEPISODIO,4.2)
  because('O valor do campo data de nascimento deve estar no formato data').

```

Fig. 4.13: Extrato de código PROLOG referente ao procedimento auxiliar de verificação da data e código FLEX de validação na regra

```

==== Predicados para agrupar em registos

agruparegisto(NUMEPISODIO):-
    findall(CODIGO,getcodigo(CODIGO,NUMEPISODIO),C),
    verifica_lista(NUMEPISODIO,C).

verifica_lista(_, []).
verifica_lista(NUMEPISODIO,[H|T]):-
    assert(registo(NUMEPISODIO,H)),
    verifica_lista(NUMEPISODIO,T).

apaga(NUMEPISODIO):-
    retractall(registo(NUMEPISODIO,_)).

```

Fig. 4.14: Extrato de código PROLOG criado para agrupar códigos do mesmo registo clínico

diagnósticos e procedimentos, assim é necessário selecionar um conjunto de linhas ao mesmo tempo.

Foi então criado um predicado para agrupar os registos clínicos, demonstrado na figura 4.14. Nesse predicado inicialmente faz-se um "find all" que seleciona e insere numa lista, todos os códigos associados a um determinado número de episódio. Posteriormente é realizada a verificação dessa lista e um "assert" para um predicado dinâmico denominado de "registo". É com este último predicado que são realizadas as validações nas regras clínicas, pois com ele é analisado código a código dentro do mesmo registo clínico.

A título de exemplo de uma regra clínica, é demonstrado na figura 4.15 a regra "cod_66" que verifica a existência de procedimentos de episiotomia simples em conjunto com procedimentos de episiotomia com o auxílio de fórceps ou ventosa. A episiotomia é uma incisão efetuada na região do períneo de forma a facilitar o parto [82]. Assim num parto só é realizado um procedimento de episiotomia, tornando inválido e por sua vez não conforme a coexistência de dois procedimentos de episiotomia no mesmo registo clínico. A regra possui as seguintes validações auxiliares e específicas:

- Predicado "apaga" de modo a eliminar qualquer argumento que o predicado dinâmico possa trazer;
- Predicado "validatecnico" de modo a associar a lista com a identificação do registo a validar;
- Validação se a regra já foi ou não disparada naquele registo clínico;

```

rule cod_66          §§511
  if apaga(NUMEPISODIO) and validatecnico(NUMEPISODIO,NOME)
  and 66 is not included in NOME
  and agruparegisto(NUMEPISODIO)
  and registo(NUMEPISODIO,L) and L is equal to 736
  and registo(NUMEPISODIO,R) and R is included in {721, 7221, 7231, 7271}
  then escreve_clinic('Episiotomia simples associada a forceps e ventosa.
  N.Episódio:',NUMEPISODIO,'Foi codificado em simultaneo um procedimento de
  episiotomia simples e de episiotomia associada a forceps ou ventosa')
  and actualiza(NUMEPISODIO,66) and apaga(NUMEPISODIO)
  because('Foi codificado em simultaneo um procedimento de episiotomia simples e de episiotomia
  associada a forceps ou ventosa').

```

Fig. 4.15: Extrato de código FLEX de validação de regras clínicas referentes a coexistência de códigos de episiotomia

- Predicado "agruparegisto" de forma a utilizar todos os códigos associados ao número de episódio e inseri-los numa lista;
- Predicado "escreve_clinic" que faz a atualização da regra na sublista da identificação do registo clínico;
- Predicado "registo" associado à validação clínica da regra, ou seja, chama a lista de códigos no registo e verifica se algum argumento é igual aos códigos, se existir o código 736 em conjunto com 721 ou 7221 ou 7231 ou 7271, a regra dispara com o alerta de não conformidade do registo clínico.

4.6.5 Construção da acção

Para correr o programa inicialmente deve ser chamado o predicado "init_bd_conn" na linha de comandos. Essa ação faz com que execute os comandos específicos para correr o DLL da *toolkit* do FLEX e do PRODATA. Também com este predicado é realizada a conexão à base de dados, e por sua vez também é assegurado a ligação dos ficheiros FLEX com o PROLOG através da ferramenta "consult".

Para correr as regras é necessário criar um ficheiro de extensão KSL. Este ficheiro, além da ação denominada de "iniciar_regras", também possui uma pequena interface gráfica realizada através da função "question" do FLEX.

Na ação é efetuado a questão, que por sua vez possui como interface uma janela *pop-up*, em que se pode definir o nome do ficheiro final. Posteriormente é executado o predicado "procuranumepi" de forma a criar as etiquetas (que no caso irão estar vazias) seguidamente é executado a função "tell" de modo a que tudo que seja *output* seja escrito no ficheiro final. Ainda nessa ação é realizado o disparar das *rulesets*

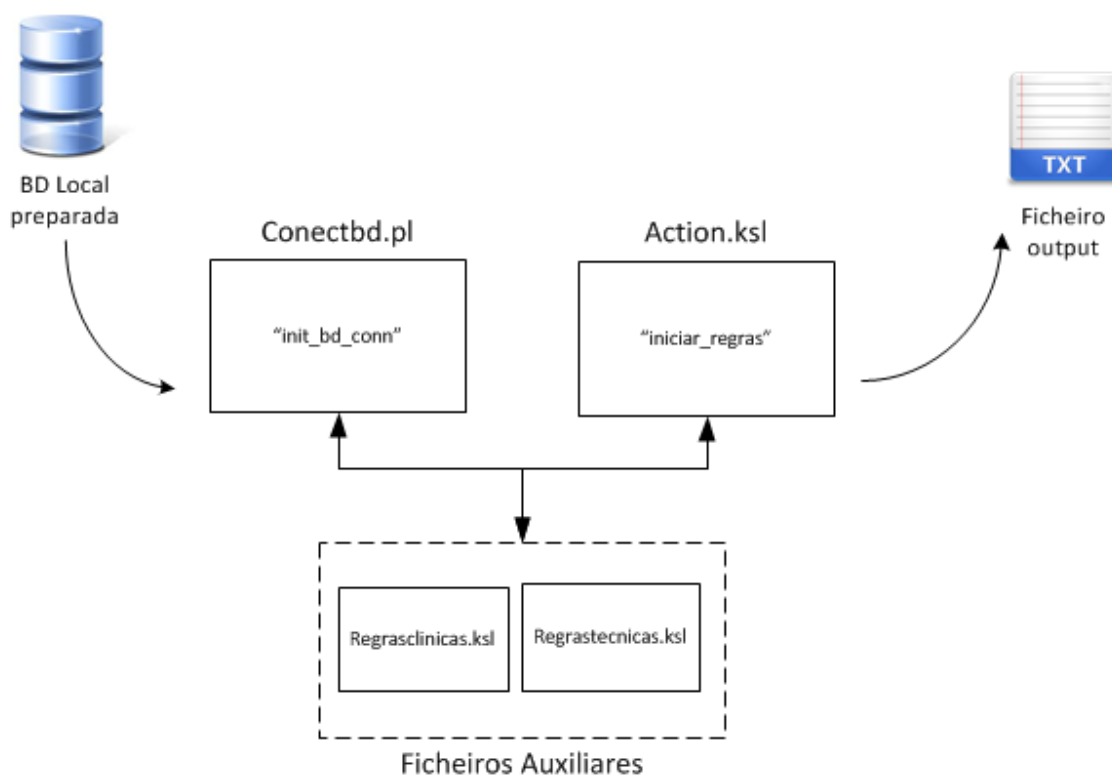


Fig. 4.16: Conexão entre ficheiros do Sistema Pericial

tanto das regras técnicas como das regras clínicas. Finalmente é executado o predicado "apaganumepi" por forma a apagar todas as etiquetas, e executado a função "told" por forma a fechar o ficheiro final.

É ainda na ação que se define a ordem das *rulesets* a serem chamadas. Assim, com as ações, é possível criar uma ligação entre todos os ficheiros tal como demonstrado na figura 4.16. Em que o ficheiro principal possui extensão PL e é executado primeiramente, o ficheiro secundário é de extensão KSL e é executado aquando de uma verificação, e finalmente os ficheiros auxiliares também de extensão KSL que contém as regras clínicas e regras técnicas.

4.7 Validação e Verificação do sistema

A verificação e a validação são processos fundamentais para a averiguação se o sistema está correto, completo e consistente [83]. Especificamente, nos Sistemas Periciais, estes dois processos são extremamente necessários, pois os conhecimentos foram adquiridos de outrem.

Pode-se dizer que a validação determina se o sistema correto foi construído, ao

Tab. 4.1: Resultados clínicos da verificação do sistema comparativamente ao Programa Auditor

Erros	Número de registos não conformes	
	Programa Auditor	Sistema Pericial
Incoerência entre o peso registado e codificado	0	1
Má posição/apresentação sem obstrução	5	5
Verificar a sequenciação dos códigos	1	0

passo que a verificação determina se o sistema foi construído corretamente [84].

A validação é normalmente um processo mais complicado, pois está dependente de uma tecnologia em particular. O’Keefe no seu artigo demonstra cinco tipos de modelos de validação: conceptual, lógico, experimental, operacional e através de dados [85]. Assim, tendo em conta os modelos referidos, pode-se considerar que o sistema construído está conceptualmente e logicamente corretos, pois o sistema está de acordo com a sua utilização, e que por sua vez descreve e soluciona corretamente a situação ou problema. Já no campo experimental e operacional o sistema não está tão adequado ao problema real. A compatibilidade do LPA PROLOG com algumas funções, exemplo do formato data, demonstrou-se bastante fraca, e de modo a contornar estes problemas foram criados procedimentos auxiliares que por sua vez diminuíram a eficiência do sistema e a operabilidade do mesmo.

A verificação foi realizada tendo em consideração os dados e tendo como fonte de comparação o programa Auditor. O objetivo dessa etapa é verificar como o sistema irá reagir ao utilizar uma amostra real.

A amostra possui aproximadamente 3900 registos num período de tempo de 6 meses. Contudo, dessa amostra apenas 230 registos foram tidos em consideração devido à especialização da área da obstetrícia, totalizando aproximadamente 2200 código associados, uma média de 10 códigos por registo.

Os resultados clínicos da verificação do sistema através de uma amostra, pode ser verificado e comparado na tabela 4.1.

É possível verificar que das três regras disparadas no conjunto, a regra que se refere a incoerência de pesos é detetado pelo SP enquanto que falha no programa Auditor. Esta regra foi verificada manualmente e por sua vez, é verdadeira. Tornando assim numa mais-valia do sistema desenvolvido comparativamente ao programa Auditor.

A segunda regra da codificação de má apresentação sem obstrução é válida tanto

no programa Auditor, como no Sistema Pericial. Ao passo que a terceira regra que verifica a sequenciação dos códigos não é apresentada no SP, pois a amostra verificada no SP é exclusivamente relacionada com a obstetrícia, fator que não ocorre no programa Auditor.

Em suma, nas regras clínicas pode-se afirmar que o sistema possui uma boa adaptação, demonstrando até validações mais corretas que o programa Auditor. Em termos de tempo gasto pode-se afirmar que o programa apesar de muito pesado possui um tempo de execução aceitável, aproximadamente 8 horas, quando comparado com todo o processo de auditoria e posterior verificação que pode demorar acima das 12 horas.

Já os resultados técnicos, são um pouco díspares, pois o sistema assume falsas não conformidades relacionadas com o número total de dias de internamento e de dias em pré-operatório. Estes erros são justificados devido à limitação da utilização do formato data em PROLOG, assim as operações com datas não tem em conta anos bissextos ou contagem de meses. O que totaliza assim três alertas errados. Uma possível solução seria a utilização de outro compilador, como por exemplo o SWI(*Sociaal-Wetenschappelijke Informatica*). O ficheiro de saída tanto do programa Auditor, como do Sistema Pericial é possível verificar nos anexos B e C, respetivamente.

4.8 Aceitação do sistema pelo utilizador

Esta etapa ainda não foi executada, contudo o piloto do sistema deveria ser testado tanto pelo perito, quanto por outros utilizadores, de modo a assegurar imparcialidade. Devido à especialização do sistema não seria possível realizar auditorias regulares, pois as amostras deverão ser seleccionadas e conter um número significativo de registos associados à obstetrícia.

Considerações Finais

Nesse capítulo serão realizadas algumas considerações globais sobre o trabalho realizado, bem como algumas sugestões a realizar numa possível continuação do mesmo.

5.1 Conclusões

Primeiramente é de realçar a dificuldade apresentada, pois a codificação é uma área bastante específica que necessita de bastante conhecimento prático, o que levou a uma procura constante ao conhecimento prático da área. Também é de realçar que o desenvolvimento de Sistemas Periciais também é uma área bastante específica que necessita de muita formação, pois a linguagem PROLOG é bastante distinta das outras linguagens de programação.

No decorrer do trabalho foram realizados diversos procedimentos paralelos, que não pertenciam ao objetivo inicial, mas que a sua possível implementação influenciava diretamente nos resultados.

O primeiro trabalho foi a implementação da aplicação para tratamento de dados, que possibilitou assim a conexão do Sistema Pericial com a Base de dados, e que num segundo ciclo de trabalhos, também possibilitou a melhoria da performance do sistema pericial.

Outro trabalho realizado, e também de extrema importância para a qualidade dos resultados, foi a transcrição de conhecimentos do raciocínio humano na área clínica para regras de validação em linguagem KSL. O grande desafio dessa transcrição era conhecer a possibilidade de implementação, visto que o raciocínio humano e principalmente na área médica têm em consideração um grande número de variáveis.

O facto de essa transcrição ser exequível, torna o sistema pericial numa ferra-

menta inovadora na área da auditoria dos registos clínicos, podendo ser implementada e adaptada a qualquer outra área médica além da obstetrícia.

O sistema pericial desenvolvido foi capaz de detetar todos os tipos de erros propostos, demonstrando ser uma ferramenta capaz de lidar com o dia-a-dia hospitalar. Ainda na verificação de resultados com uma determinada amostra real, o sistema ainda foi capaz de ultrapassar o atual programa em utilização - Programa Auditor, num erro de carácter administrativo com posterior influência na aplicação do GDH daquele registo. O sistema é ainda capaz de fornecer explicações o que torna a verificação menos morosa e aumenta a confiança entre aplicação e auditor.

Através do disposto anteriormente, é possível então verificar que o SP desenvolvido, apesar de ser ainda um protótipo e aplicado a uma área restrita, demonstra qualidade em detetar não conformidades nos registos clínicos, que posteriormente traduzem-se numa correta taxa de produção, numa faturação conforme, numa produção de case-mix apurada, que por sua vez irá traduzir num modelo de financiamento mais justo para o Hospital. Numa perspectiva de qualidade traduzir-se-á numa qualidade de todos os elementos do sistema e consequente qualidade nos serviços de saúde prestados.

5.2 Limitações e Trabalho futuro

Ao longo do trabalho, alguns aspectos ficaram aquém do planeado inicialmente, como por exemplo, o facto de o compilador utilizado (LPA) não reconhecer o formato data, o que se tornou num processo mais demorado e com inconsistências. Na verificação de resultados, esse aspecto tornou-se claro, evidenciando falsas não conformidades.

Comparativamente com outro compilador, por exemplo o SWI PROLOG, o formato data torna-se tão simples como declarar que o atributo é do tipo "date". Ainda no SWI PROLOG é permitido operações com o formato através de predicados específicos como o "day_of_the_week" ou "get_date". Contudo o LPA PROLOG é a ferramenta lecionada no Mestrado, e que por sua vez possuía algumas noções prévias ao início da Tese.

Outro aspecto também de realçar é os procedimentos adicionais para a realização de ciclos, no caso em particular da criação das etiquetas. Aquando da implementação do sistema diversas técnicas foram testadas por forma a auxiliar a *ruleset* para a criação do ciclo, contudo somente o procedimento utilizando listas e sublistas operava corretamente. Esse procedimento por sua vez, também influenciava o sistema tornando-o mais pesado e consequentemente mais lento.

Outra limitação foi a amostra utilizada para a realização de testes no programa desenvolvido. O facto de esta ser pequena (comparativamente com a realidade da base de dados hospitalar), com poucos erros e ter de ser orientada para uma área em específico, leva a que a validação das regras/sistema sejam incertas devido a sua realização num ambiente controlado. Contudo e aquando do desenvolvimento foram realizadas manualmente alteração e modificações pontuais em dados de modo a que o tipo de erros encontrados fosse mais alargado, criando um acréscimo na validação do sistema.

A nível de trabalhos futuros, seria interessante abranger outras áreas médicas, evitando assim a seleção de registos. Num nível mais comercial, e caso o objetivo passa pela comercialização, ao invés apenas da investigação, era pertinente a alteração de algumas componentes, como por exemplo o compilador PROLOG. Assim evitava-se alguns procedimentos auxiliares, tornando sem dúvida numa ferramenta mais rápida. Outros trabalhos complementares seria a construção de uma interface gráfica amigável ao utilizador, a implementação de um editor de regras, possibilitando assim uma alteração mais simples ou até mesmo a inserção de novas regras, ou até mesmo a implementação de multi-utilizadores na aplicação.

Bibliografia

- [1] IPAI, “Auditoria interna.” URL: <http://www.ipai.pt>, Maio 2013.
- [2] S. P. Frostick, P. J. Radford, and W. A. Wallace, “Introduction,” in *Medical Audit: Rationale and Practicalities*, Cambridge University Press, 1993.
- [3] A. S. Barreto, N. Amaro, T. Boto, and J. Marques, “Auditorias à codificação clínica - metodologia e apresentação dos resultados,” tech. rep., Ministério da Saúde - ACSS, 2010.
- [4] J. Ivreitveit and P. Sousa, *Quality Safety improvement research: methods and research practice from the international quality improvement research network (QIRN)*. Affanumérico Lda, 2008.
- [5] A. Freire, *Gestão da doença e qualidade em saúde*, ch. Estratégia e Qualidade. Comissão Sectorial da Saúde do IPQ, Dezembro 2000.
- [6] R. Pinto, “Sistemas de informações hospitalares do brasil, espanha e portugal - semelhanças e diferenças,” Master’s thesis, Escola Nacional de Saúde Pública Sérgio Arouca, Janeiro 2010.
- [7] F. Lopes, “Portal da codificação e dos gdh: A codificação clínica.” URL: http://portalcodgdh.min-saude.pt/index.php/Codificação_clínica, Junho 2013.
- [8] H. Quan, B. Li, L. D. Saunders, G. A. Parsons, C. I. Nilsson, A. Alibhai, and W. A. Ghali, “Assessing validity of icd-9-cm and icd-10 administrative data in recording clinical conditions in a unique dually coded database,” *Health Research and Educational Trust*, pp. 1424–1441, 2007.
- [9] W. E. Hammond and J. J. Cimino, *Biomedical Informatics*, ch. Standards in Medical Informatics, pp. 265–311. Springer, 2006.
- [10] F. Lopes, “Portal da codificação e dos gdh: Classificação internacional de doenças (cid).” URL: <http://portalcodgdh-min-saude.pt/index.php/Cid>, Agosto 2013.

- [11] A. C. Hart, M. S. Stegman, and B. Ford, *ICD-9-CM for Hospitals*. INGENIX, 1 ed., 2010.
- [12] AMA, “The differences between icd-9 and icd-10,” tech. rep., AMA - American Medical Association, June 2010.
- [13] G. JIang, I. Pathak, and C. G. Chute, “Formalizing icd coding rules using formal concept analysis,” *Journal of Biomedical informatics*, vol. 42, pp. 504–517, 2009.
- [14] A. Hazlewood, “Icd-9-cm to icd-10: Implementation issues and challenges,” in *AHIMA’s 75th Anniversary National Convention and Exhibit Proceedings*, AHIMA, October 2003.
- [15] C. M. Borges, “Portal da codificação e dos gdh: Grupo de diagnóstico homogéneo (gdh).” URL: <http://portalcodgdh.min-saude.pt/index.php/GDH>, Junho 2013.
- [16] R. Averell, R. Mullina, P. Giardi, and E. Elia, *Diagnosis Related Groups - Definitions Manual*. Health Systems International, 1985.
- [17] J. Urbano and M. Bentes, “Definição da produção do hospital : os grupos de diagnósticos homogéneos.” Conferência sobre o financiamento e gestão de serviços hospitalares, 1988.
- [18] M. F. Noronha, C. T. Veras, I. C. Leite, M. S. Martins, F. B. Neto, and L. Silver, “The development of diagnosis related groups - drg’s a methodology for classifying hospital patients,” *Revista Saúde Pública*, vol. 25, no. 3, pp. 198 – 208, 1991.
- [19] F. Lopes, “Portal da codificação e dos gdh: Implementação dos gdh em portugal - os primeiros anos.” URL: http://portalcodgdh.min-saude.pt/index.php/Implementação_dos_GDH_em_Portugal:_os_primeiros_anos, Junho 2013.
- [20] C. E. Dsimuke and V. Sena, “Has drg payment influenced the technical efficiency and productivity of diagnostic technologies in portuguese public hospitals? an empirical analysis using parametric and non.parametric methods,” *Health Care Management science*, vol. 2, pp. 107–116, 1999.
- [21] M. Bentes, M. L. Gonçalves, S. Tranquada, and J. Urbano, “A utilização dos gdh como instrumento de financiamento hospitalar,” *Gestão Hospitalar*, pp. 33–42, 1996.
- [22] C. M. Borges and F. Lopes, “Portal da codificação e dos gdh: Peso relativo de um gdh.” URL: http://portalcodgdh.min-saude.pt/index.php/Peso_relativo_dum_GDH, Junho 2013.
- [23] C. Costa and S. Lopes, “Produção hospitalar: a importância da complexidade e da gravidade,” *Revista Portuguesa de Saúde Pública*, vol. 4, pp. 35–50, 2004.

- [24] C. M. Borges, "Portal da codificação e dos gdh: Índice de case-mix." URL: [http://portalcodgdh.min-saude.pt/index.php/Índice_de_Case-Mix_\(ICM\)](http://portalcodgdh.min-saude.pt/index.php/Índice_de_Case-Mix_(ICM)), Junho 2013.
- [25] F. George, J. L. Castanheira, J. N. Martins, A. R. Larangeira, B. Rodrigues, and T. M. de los Rios, "Health in portugal 2007," tech. rep., Directorate - General of health - Ministry of health, July 2007.
- [26] T. Lehtonen, "Drg-based prospective pricing and case-mix accounting - exploring the mechanisms of succesful implementation," *Management Accounting Research*, vol. 18, pp. 267–395, 2007.
- [27] R. Braga, "Os registos clínicos e a codificação," *Rev. Port. Med. Geral Fam.*, vol. 28, pp. 155–156, 2012.
- [28] H. Francis and R. France, "Case mix in 25 countries: a migration sucess but international comparisons failure," *International Journal of medical informatics*, vol. 70, pp. 215–219, 2003.
- [29] M. Bentes, M. do Céu Mateus, and M. da Luz Gonsalves, "Drgs in portugal. a decade of experience," tech. rep., Ministério da Saúde - IGIF, 2000.
- [30] J. Price, C. A. Estrada, and D. Thompson, "Administrative data versus corrected administrative data," *American Journal of Medical Quality*, vol. 18, no. 1, pp. 38–45, 2003.
- [31] T. N. da Silva e Costa, "Indicadores de produção hospitalar. uma forma de medir a produção dos hospitais portugueses," Master's thesis, Faculdade de Medicina da Universidade do Porto, Junho 2010.
- [32] B. Steinwald and L. A. Dummit, "Hospital case-mix change: sicker patients or drg creep?," *Health Affairs*, vol. 8, pp. 35–47, 1989.
- [33] E. D. Gomes, A. F. de Araújo, and R. J. Barbosa, "Auditoria: Alguns aspectos a respeito da sua origem," *Revista Científica electrónica de ciências contábeis*, vol. 13, Maio 2009.
- [34] P. J. Costello, "Auditing concepts and standards," Tech. Rep. 6, NPMA, 2003.
- [35] A. Services, "Continuous auditing," Tech. Rep. EB/CA/092706, ACL ltd, 2006.
- [36] S. Aamir, "Auditor client relationship and audit quality," Master's thesis, Unea School of Business, 2011.
- [37] A. W. Ramos, *Auditorias da qualidade*, vol. 1. Produção, 2 ed., 1991.
- [38] J. O. Imoniana, *Auditoria de sistemas de Informação*. Atlas, 2 ed., 2005.
- [39] N. S. Antonio and A. Teixeira, *Gestão da qualidade - de Deming ao modelo de excelência do EFQM*. 1 ed., 2007 2007.

- [40] D. R. Bamford and R. W. Greatbanks, "The use of quality management tools and techniques: a study of applications in everyday situations," *International Journal of Quality and Reliability Management*, vol. 22, no. 4, pp. 376–392, 2005.
- [41] S. Manson, S. McCartney, and M. Sherer, "Audit automation as control within audit firms," *Journal of Accounting, Finance and Management*, vol. 14, pp. 109–130, 2001.
- [42] M. Sherer and S. Manson, "Audit automation. improving quality or keeping up appearances?," in *Current Issues in Auditing*, ch. 14, pp. ISBN: 1–85396–365–8, SAGE, 1997.
- [43] M. R. Lyra, *Segurança e Auditoria em Sistemas de Informação*. Ciência Moderna, 2008.
- [44] E. C. Terruel, "Principais ferramentas utilizadas na auditoria de sistemas e as suas características," 2010.
- [45] T. Singleton, "Generalized audit software: Effective and efficient tool for toda's it audits," tech. rep., ISACA, 2006.
- [46] A. Mchoes and I. M. Flynn, *Understanding operating systems*, ch. Chapter 14 - MS-DOS operating system. Cengage Learning, 6 ed., 2010.
- [47] IGIF, "Manual do programa auditor v3.18," tech. rep., IGIF - Instituto de gestão informática e financeira da saúde, Novembro 2005.
- [48] L. Spector, "Evolution of artificial intelligence," tech. rep., 2006.
- [49] P. Kumar, "Artificial intelligence and fuzzy logic controller in a plc system." Electrical Engineering and Power Electronics - 9th Research Seminar Series Workshop, April 2010.
- [50] K. Omoteso, "The application of artificial intelligence in auditing: looking back to the future," *Expert Systems with Applications*, vol. 39, pp. 8490 – 8495, 2012.
- [51] B. K. Wong and J. A. Monaco, "A bibliography of expert systems applications for business (1984-1992)," *European Journal of Operational Research*, vol. 85, pp. 416–432, 1995.
- [52] P. Jackson, *Introduction to Expert Systems*. Addison-Wesley, 3rd ed., 1999.
- [53] J. Alves, "Avaliação de desempenho dos sistemas periciais," Master's thesis, Universidade Técnica de Lisboa - Instituto Superior de Economia e Gestão, Julho 2005.
- [54] L. R. Ye, "The value of explanation in expert systems for auditing: an experimental investigation," *Expert Systems and Applications*, vol. 9, no. 4, pp. 543–556, 1995.

- [55] K. Erik Arzén, “An architecture for expert system based in feedback control,” *Automanca*, vol. 25, no. 6, pp. 813–827, 1989.
- [56] L. R. dos Santos, “Protótipo de um sistema educativo para diagnóstico médico,” Master’s thesis, Centro Universitário Vila Velha, 2010.
- [57] I. Kuo and A. Mita, “Quality control expert systems: a review of pertinent literature,” *Journal of Intelligent Manufacturing*, vol. 4, pp. 245 – 257, 1993.
- [58] E. P. Paladini, “An expert systems approach to quality control,” *Expert Systems with Applications*, vol. 18, pp. 133 – 151, 2000.
- [59] Y. Goldberg and A. Shmilovici, “An expert system approach for quality assurance auditing,” *J. Adv. Manuf. Technol*, vol. 26, pp. 413 – 419, 2005.
- [60] M. A. Vasarhelyi and A. Kogan, *Artificial Intelligence in Accounting and Auditing: Towards new Paradigms*, vol. 4. Markus Wiener Publisher, 2000.
- [61] D.-E. Lee, T.-K. Lim, and D. Arditi, “An expert system for auditing quality management systems in construction,” *Computer Aided Civil and Infrastructure Engineering*, vol. 26, pp. 612 – 631, 2011.
- [62] R. Moore, H. Rosenof, and G. Stanley, “Process control using a real time expert system,” tech. rep., IPAC, 1990.
- [63] A. S. Tomas, *Artificial Intelligence in Accounting and Auditing: Towards new paradigms*, vol. 4, ch. Surveys - Expert Systems in Auditing. Markus Wiener Publisher.
- [64] W. P. Wagner, J. Otto, and Q. B. Chung, “Knowledge acquisition for expert systems in accounting and financial problems domain,” *Knowledge-Based Systems*, vol. 15, pp. 439 – 447, 2002.
- [65] W. V. Melle and E. H. Shortliffe, “What is medical artificial intelligence? - emycin: A knowledgeengineer’s tools for constructing rule-based expert systems.”
- [66] W. J. Claney and E. A. Shortliffe, “What is medical artificial intelligence? - introduction: Medical artificial intelligence programs.”
- [67] J. Wyatt and D. Spiegelhalter, *Knowledge Based Systems in Medicine: Methods, applications and evaluation lectures notes in Medical Informatics*, vol. 47, ch. Evaluating Medical Expert Systems: what to test, and how?, pp. 274 – 290. 1991.
- [68] A. Wright and D. F. Sittig, “Sands: A service-oriented architecture for clinical decision support in national health information network,” *Journal of Biomedical Informatics*, vol. 41, pp. 962 – 981, 2008.

- [69] T. Mabotuwana and J. Warren, "Chronomedit: A computacional quality audit framework for better management of patients with chronic conditions," *Journal of Biomedical Informatics*, vol. 43, pp. 144 – 158, 2010.
- [70] D. Bayraktar, "A knowledge-based expert system approach for the auditing process of some elements in quality assurance system,," *Int. J. Production Economics*, pp. 37–46, 1998.
- [71] H. EPE, "Relatório de contas 2011," tech. rep., Centro Hospitalar São João EPE, 2011.
- [72] J. M. Siqueira and V. Rocio, "Sistema pericial em prolog no diagnóstico potencial agrícola de solos." VI Congresso Ibérico de Agro Engenharia, 2011.
- [73] W. R. Swartout and S. W. Smoliar, "On making expert systems more like experts," *Expert Systems*, vol. 4, pp. 196–208, 1987.
- [74] C. Spencer, "Flex tutorial," tech. rep., Logic Programming Associates Ltd, June 2004.
- [75] R. Lucas, "Prodata interface," tech. rep., Logic Programming Associates Ltd, June 2004.
- [76] R. Shalfield, "User guide lpa prolog," tech. rep., Logic Programming Associates Ltd, September 2008.
- [77] C. L. Wiggs and R. S. Perez, "The use of knowledge acquisition in instructional design," *Computers in Human Behavior*, vol. 4, no. 3, pp. 257–274, 1988.
- [78] V. Y. Yoon and M. Adya, *Encyclopedia of information systems*, ch. Expert systems construction, pp. 291–300. Elsevier, 2003.
- [79] R. Lessa and E. Junior, "Modelo de processos de engenharia de software de software." UNISUL - Princípios da Engenharia de Software, 2003.
- [80] M. Soares, "Metodologias ágeis: Extreme programming e scrum para o desenvolvimento de software." Brasil, 2004.
- [81] F. Cismandi, A. S. Fialho, S. M. Vieira, S. R. Reti, J. M. C. Sousa, and S. N. Finkelstein, "Missing data in medical data bases: impute, delete or classify?," *Journal of Artificial Intelligence in Medicine*, vol. 58, pp. 63–72, 2013.
- [82] R. Mattar, M. Aquino, and M. Mesquita, "A prática doa episiotomia no brasil," *Revista Brasileira de Ginecologia e Obstetrícia*, vol. 29, no. 1, pp. 1–2, 2007.
- [83] C. Anantaram, G. Nagaraja, and K. V. Nori, "Verification of accuracy of rules in a rule based systems," *Data and Knowledge engineering*, vol. 27, pp. 115–138, 1998.

-
- [84] S. Lockwood and Z. Che, “Knowledge validation of engineering expert systems,” *Advances in Engineering Software*, vol. 29, pp. 97–104, 2007.
- [85] R. O’Keefe and D. O’Leary, “Expert system verification and validation: a survey and tutorial,” *Artificial Intelligence Review*, vol. 7, pp. 3–42, 1993.

Apêndice **A**

Tabela de regras técnicas e clínicas

Nº	Nome Regra	Descrição
1	admin_1	Código de Hospital não preenchido
2	admin_1_2	Código de Hospital inválido (diferente da tabela de identificação dos Hospitais)
3	admin_2	Número de doente não preenchido ou inválido (valor negativo)
4	admin_3	Sexo não preenchido ou inválido (diferente de 1,2 ou 3)
5	admin_4	Campo data de nascimento não preenchido
6-8	admin_4_2 - admin_4_4	Campo data de nascimento num formato incorrecto (validação dia,mês e ano)
9	admin_5	Entidade financeira responsável não preenchido ou inválido (diferente de 6 caracteres)
10	admin_7	Destino após alta não preenchido ou inválido (ser negativo e diferente de 1,2,6,7,13,20,51,61 e 63)
11	admin_10	Campo tipo de admissão não preenchido ou inválido (negativo e diferente de 1 a 7)
12	admin_11	Hospital de destino não preenchido (somente se DSP=2)
13	admin_11_2	Hospital de destino inválido (diferente da tabela de identificação dos Hospitais)
14	admin_12	Hospital de proveniência não preenchido (somente se PROVEN=9)
15	admin_12_2	Hospital de proveniência inválido (diferente da tabela de identificação dos Hospitais)
16	admin_13	Médico responsável não preenchido ou inválido (diferente de 5 caracteres)
17	admin_14	Campo data de alta não preenchido
18-20	admin_14_2 - admin_14_4	Campo data de alta num formato incorrecto (validação dia,mês e ano)
21	admin_35	Campo data de admissão não preenchido
22-24	admin_35_2 - admin_35_4	Campo data de admissão num formato incorrecto (validação dia,mês e ano)
25-36	admin_15_1 - admin_15_12	Total de dias incorrecto (caso seja diferente de SAIDLAST - DATAADMISSAO)
37	admin_15_13	Total de dias de internamento não preenchido ou inválido (valor negativo)
38	admin_9	Campo ICU inválido (Caso esteja preenchido e se [CU > TOTDIAS)
39-42	admin_16_1 - admin_16_4	Campo idade incorrecta (caso seja diferente de DATAADMISSAO - BDATE)
43	admin_16_5	Campo idade não preenchida ou inválida (valor negativo ou acima de 124)
44	admin_26	Campo inválido (negativo e acima de 22)
45-66	admin_27_1 - admin_27_22	Campo inválido (negativo e acima de um determinado valor - consoante o distrito em questão)
67	admin_28	Campo inválido (negativo e acima de 50)
68	admin_29	Número de episódio não preenchido ou inválido (valor negativo)
69	admin_30	Data de intervenção cirúrgica inferior a data de admissão
70-72	admin_30_2 - admin_30_4	Campo data da intervenção cirúrgica num formato incorrecto (validação dia,mês e ano)
73	admin_14_5	Data de alta inferior a data de admissão

Nº	Nome Regra	Descrição
74-85	admin_31_1 - admin_31_12	Campo pré-operatório incorrecto (diferente de INTERV_CIR - DATAADMISAO)
86	admin_32	Médico codificador não preenchido ou inválido (diferente de 5 caracteres)
87-95	admin_33_1 - admin_33_9	Destino após alta GDH inválido (consoante o preenchimento de DSP)
96	admin_34	Motivo de transferência inválido (se for um valor negativo ou superior a 4 - somente se DSP = 2)
97	admin_36	Data de exportação do serviço não preenchida
98	admin_37	Módulo não preenchido ou inválido (se for diferente de INT, CAMB ou AMBM)
99	admin_40	Campo hora de admissão não preenchido
100	admin_41	Campo hora de alta não preenchido
101	admin_46	Proveniência não preenchido ou inválido (valor negativo ou superior a 11)
102	admin_52	Campo diagnóstico principal não preenchido
103	admin_54	Campo serviço alta não preenchido
104	admin_55	Campo numero de episódio hospitalar não preenchido
105-131	admin_17_1 - admin_17_27	Grupo de diagnóstico homogêneo inválido (consoante o DL 839-A-2009 e de 2013)
132	admin_18	Grande categoria de diagnóstico inválida (consoante o DL 839-A-2009 e de 2013)
133-135	cod_62_1 - cod_62_3	Complicações de abortamento em codificação múltipla (erro:297)
136	cod_63	Diagnóstico com parto referenciado, mas procedimento não codificado (erro:378)
137	cod_64	Diagnóstico de parto por forceps ou ventosa e procedimento não codificado (erro:439)
138	cod_65	Episiotomia sem codificação do procedimento de parto (erro:504)
139	cod_66	Episiotomia simples e associada a forceps ou ventosa (erro:511)
140	cod_67	GDH469 - Diagnóstico principal inválido como diagnóstico de alta (erro:574)
141-142	cod_75_1 - cod_75_2	Gestação Múltipla: verificar o diagnóstico do produto de parto (erro:570)
143	cod_69_2	Imaturidade extrema: verificar a não codificação das semanas de gestação (erro:621)
144-152	cod_58_1 - cod_58_9	Incoerência entre os pesos registados na DB e codificado na CID-9 (erro:634)
153	cod_76	Ma nutrição fetal: verificar a não codificação das semanas de gestação (erro:717)
154	cod_69_1	Prematuridade: verificar a não codificação das semanas de gestação
155-156	cod_59_1 - cod_59_2	Peso à nascença inválido (erro:770)
157	cod_70_1	Registo com parto referenciado, diagnóstico com 5º dígito incoerente (erro:812)
158	cod_70_2	Registo com parto referenciado, falta o diagnóstico e o produto (erro:819)
159	cod_70_3	Registo com parto referenciado, falta procedimento e o produto (erro:823)
160	cod_70_4	Registo com parto referenciado, falta o produto do parto (erro:826)

Apêndice **B**

Ficheiro Output do Programa Auditor

Mensagens do aparelho circulatório	nº	%	confirm. %	leg.
Angina de peito associada a diagnóstico de EAM				(99)
Aterosclerose extrem. e gangrena em codif. múltipla				(99)
AVC assoc. a doença vasc. cerebral mais específica	1	1.39		(99)
Circulação extracorporal e hipot./plegia/PM per-op.				(99)
Combinação errada de pacemaker e terminais				(14)
Inserção de stent coronário s/angioplastia coronár.	1	1.39		(99)
Inserção/remoção pacemaker s/terminais ou vice-vers	1	1.39		(99)
Doença mitroaórtica em codificação múltipla				(99)
Doença cardíaca hipertensiva em codificação múltipl	5	6.94		(99)
Insuficiência cardíaca em codificação múltipla				(99)
Insuficiência cardíaca esq. associada a insuf. dta.				(99)
Códig. simult. presença e ausência de insuf. card.				(99)
Códig. simult. presença e ausência enfarte cerebral				(99)
Síndrome coronário intermediário associado a EAM	1	1.39		(99)
Verificar possível dç card hipertensiva em cod múlt				(99)
Verificar sequenciação síndr.coronár. intermediário				(99)
Verificar n/codif dça valvular em cirurgia valvular				(99)
Verificar n/codificação de CEC em cirurgia valvular	1	2.13		(99)
Mensagens de Obstetrícia / Recém-Nascidos	nº	%	confirm. %	leg.
Código de maternidade no registo do RN				(19)
Código de Recém-Nascido no registo da mãe				(18)
Códig. simult. presença e ausência prolapso uterino				(99)
Códig. simult. prolapso vaginal completo e incompl.				(99)
Verificar Diag.P. em Adulto c/ GDH de R/N (385-391)				(99)
Peso preenchido em registo de "não Recém-Nascido"				(99)
Incoerência entre o peso regist. na BD e peso codif				(99)
Prematuridade: verificar o peso registado				(99)
Prematuridade: verificar a n/codif.semanas gestação				(99)
Imaturidade extrema: verificar o peso registado				(99)
Imaturidade extr.: verif. n/codif. semanas gestação				(99)
Malnutrição fetal: verif. n/codif. semanas gestação				(99)
Diagnóst. c/parto referenciado mas parto n/codific.				(18)
Diagn. parto por fórceps/ventosa s/ proc.respectivo				(18)
Registo c/parto refer., diagn. c/ 5º díg.incoerente				(18)
Registo c/parto referenc., falta o produto do parto				(18)
Gestação múltipla: verificar diagn.produto do parto				(18)
Registo c/parto referenc., falta proced. e produto				(18)
Registo c/parto referenc., falta diagn. e produto				(18)
Registo c/produto parto, falta diagnóst. e proced.				(18)
Episiotomia s/codificação do procedimento do parto				(15)
Episiotomia simples (e episiot.) assoc.fórceps/vent				(15)
Parto referenc. em mulher s/diagnóstico maternidade				(18)
Parto normal e complicado simultaneamente				(18)
Verif. codif. parto assistido e p/cesariana simult.				(18)
Verif. codif. má posição/apresentação sem obstrução	5	7.25		(18)
Verificar sequenciação dos códigos de obstetrícia	1	1.45		(18)
Verif. sequenc. dos cód. de obstrução e má apresent				(18)

Apêndice **C**

Ficheiro Output do Sistema Pericial

auditoria3.txt

----- Regras Técnicas -----

Numero total de dias de internamento incorrecto. N. Paciente: 13010842
N. Episódio: 13013381
Verifique se o numero total de dias corresponde a subtração da data de saída com a data de admissão

Numero total de dias de internamento incorrecto. N. Paciente: 12031263
N. Episódio: 12035068
Verifique se o numero total de dias corresponde a subtração da data de saída com a data de admissão

Dias de Pré-operatório incorrecto. N. Paciente: 7016643 N. Episódio: 13013727
Verifique se o numero de dias em pré-operatório corresponde a subtração da data da primeira intervenção cirurgica e a data de admissão

Dias de Pré-operatório incorrecto. N. Paciente: 12031263 N. Episódio: 12035068
Verifique se o numero de dias em pré-operatório corresponde a subtração da data da primeira intervenção cirurgica e a data de admissão

----- Regras Clínicas -----

Incoerência entre o peso registo e o peso codificado. N. Episódio: 13009440
Foi codificado um diagnóstico da gama 764XX, 7650X e 7651X com quinto dígito 1, que normalmente implica que o peso deve estar compreendido entre 300 e 500g

Verificar a não codificação da obstrução em má posição/apresentação.
N. Episódio: 13015644
No registo em análise foi codificado uma má posição ou má apresentação, mas não a possível obstrução do parto.

Verificar a não codificação da obstrução em má posição/apresentação.
N. Episódio: 13015641
No registo em análise foi codificado uma má posição ou má apresentação, mas não a possível obstrução do parto.

Verificar a não codificação da obstrução em má posição/apresentação.
N. Episódio: 13015547
No registo em análise foi codificado uma má posição ou má apresentação, mas não a possível obstrução do parto.

Verificar a não codificação da obstrução em má posição/apresentação.
N. Episódio: 13015296
No registo em análise foi codificado uma má posição ou má apresentação, mas não a possível obstrução do parto.

Verificar a não codificação da obstrução em má posição/apresentação.
N. Episódio: 13013991
No registo em análise foi codificado uma má posição ou má apresentação, mas não a possível obstrução do parto.

--- Terminado ---

