

# DESENVOLVIMENTO DE APLICAÇÃO PARA COMUNICAÇÃO E ARMAZENAMENTO DE IMAGENS MÉDICAS DIGITAIS EM BANCO DE DADOS

Nelson D. Evangelista<sup>1</sup>, Juliana F. Camapum<sup>2</sup>

**Resumo.** O trabalho tem como principal objetivo o desenvolvimento de um projeto protótipo de comunicação e arquivamento de imagens clínicas, mais conhecidas mundialmente como PACS (*Picture Archiving Communication System*), utilizando o protocolo DICOM (*Digital Imaging and Communications in Medicine*), desenvolvido por meio da cooperação entre a indústria de equipamentos médicos e a comunidade de usuários. O padrão estabelece uma linguagem comum que permite que imagens médicas e informações de um determinado equipamento estejam disponíveis em meio digital através da interconexão dos dispositivos em redes padronizadas. O artigo apresenta a implementação de uma aplicação composta por três servidores de banco de dados distintos e um *software* cliente DICOM, que utiliza o protocolo TCP/IP, com o intuito de prover os serviços de comunicação, transmissão, armazenamento e gerenciamento de imagens médicas. A implementação do servidor de imagens considera a adoção do padrão DICOM, que não apenas define o formato de armazenamento, mas também o protocolo de comunicação entre o servidor e as estações clientes. As principais contribuições do trabalho estão na obtenção de novos conhecimentos de tecnologias de arquivamento e comunicação de imagens que será utilizado pelos hospitais e clínicas médicas para facilitar a aquisição e arquivamento de um grande acervo de imagens médico-hospitalares resultando em um acréscimo significativo de eficiência, produtividade e qualidade destas organizações.

Palavras-chave: PACS. DICOM. Banco de dados.

**Abstract.** The paper has as its essential purpose the development of a prototype project of communication and storage of clinical images, globally known as PACS (Picture Archiving Communication System), using the protocol DICOM (Digital Imaging and Communications in Medicine), developed thanks to the cooperation between the medical equipment industry and the users' community. The standard establishes a common language which allows clinical images and information from determined equipment to be available in digital means through the interconnection of the devices in standardized webs. The project presents the implementation of an application composed by three servers of different databases and a DICOM customer

<sup>1</sup> Mestrado em Engenharia de Sistemas e Computação. Escola de Administração do Exército (EsAEx), Salvador, Brasil. nelsonnev@brturbo.com.br.

<sup>2</sup> Doutorado em Engenharia Elétrica. Universidade de Brasília (UnB), Brasília, Brasil. juliana@ene.unb.br.

software, which uses the TCP/IP protocol, so as to provide the communication, transmission, storage and management of medical images services. The implementation of the image server considers the adoption of the DICOM standard. It doesn't only define the storage format, but also the communication protocol between the server and the customer stations. The main contributions of the paper are the acquisition of new knowledge about image database and communication technologies which will be used by the hospitals and medical clinics to facilitate the acquisition and storage of a large medical image database created by doctors. This will result in a significant increment of efficiency, productivity and quality of these organizations.

Keywords: PACS. DICOM. Database.

## 1 Introdução

O expressivo progresso da radiologia nas últimas décadas, associado ao aparecimento e aperfeiçoamento de novos métodos de imagem, fez do diagnóstico por imagem uma das mais excitantes áreas da Medicina. Seu impacto é tão grande que a abordagem diagnóstica médica vem sofrendo sua influência e sendo bastante modificada. Dentre as várias modificações, a implantação dos Sistemas de Comunicação e Arquivamento de Imagens - o PACS (*Picture Archiving and Communication System*) é provavelmente a de maior impacto (ALMEIDA, 1998).

Esta nova realidade motiva este trabalho a buscar o conhecimento do padrão DICOM, prover uma aplicação capaz de realizar a captura destes pacotes de mensagens e o seu armazenamento em um sistema

gerenciador de banco de dados e o armazenamento desta imagem em disco rígido.

Este trabalho apresenta a implementação e uma aplicação composta por três servidores de banco de dados distintos (*Firebird, PostgreSQL e Oracle*) e um *software* cliente DICOM.

Tal conjunto, disponível como protótipo, faz uso do protocolo TCP/IP com o intuito de prover os serviços de comunicação e transmissão. Todas as operações realizadas por este conjunto de aplicações são executadas em conformidade com a especificação do padrão.

## 2 Protocolo DICOM

O padrão DICOM é um conjunto de regras que permite que as imagens médicas e suas informações associadas sejam transmitidas entre equipas

mentos de imagem, computadores e hospitais. Entretanto, antes de iniciar a conexão de diversos equipamentos em uma rede DICOM de comunicação é necessário que a Declaração de Conformidade DICOM (*DICOM Conformance Statement*) de cada equipamento seja analisado de forma a avaliar se são realmente compatíveis (DICOM, 2003).

## 2.1 Constituição do padrão DICOM

A especificação do padrão DICOM 3.0 encontra-se dividida em várias partes. Tal divisão permite que cada parte possa expandir-se, individualmente, sem haver a necessidade de reeditar todo o padrão. Dentro das partes as seções se sujeitam a adições ou alterações reduzindo assim o esforço de edição necessário quando ocorre a atualização das mesmas (DICOM, 2003).

## 2.2 Formato do arquivo do padrão DICOM

Foi descrito com base na Parte 10, que define o formato de arquivos e armazenamento em mídia (*Media storage and file format*). Tal formato proporciona uma maneira de encapsulamento em um arquivo dos

Conjuntos de Dados representados em uma Instância SOP relativo a um determinado IOD (*Information Object Definition*). O conjunto de *bytes* do *Data Set* é colocado em um arquivo após o cabeçalho designado pelo padrão *DICOM File Meta Information Header*. Cada arquivo contém apenas um único Objeto de Informação (*Information Object*) / Instância SOP (*SOP Instance*).

## 2.3 Cabeçalho

Este cabeçalho inclui a informação de identificação do *Data Set* encapsulado. É constituído por um preâmbulo (*File Preamble*) de 128 (cento e vinte e oito) *bytes*, seguido de um prefixo DICOM de quatro *bytes*, que por sua vez é seguido pelos Elementos do Arquivo de Meta Dados (*File Meta Elements*). Este cabeçalho é obrigatório em todos os arquivos padrão DICOM.

Com as exceções do preâmbulo e do prefixo “DICM”, todo o resto do cabeçalho deve ser codificado usando a Sintaxe de Transferência. Optou-se por usar a notação hexadecimal para representar todos os números inteiros apresentados em seguida por ser esta a notação convencional nas especificações do padrão DICOM. A letra “H” posterior a um número, sig-

nifica que este está representada em Hexadecimal. Apesar de ser omitido esse “H” nas Etiquetas / Rótulos (*Tag*) dos Elementos de Dados todas elas estão também representadas em Hexadecimal. Dessa forma, uma Etiqueta / Rótulo (*Tag*) é representada(o) como (gggg,eeee) , sendo “gggg” o Número do Grupo (*Group Number*) e “eeee” o Número do Elemento (*Element Number*) dentro deste Grupo, sendo ambos representados em formato hexadecimal. Todas as Etiquetas / Rótulos (*Tag*) que possuem este formato (0002,eeee) estão reservados

para este padrão e para versões anteriores ao DICOM.

É observado na figura 1 o arquivo com um preâmbulo com 128 bytes, os caracteres “DICM”, a representação dos Grupos/Elementos na representação *Little Endian* (0002, 0001; 0002,0002; 0002,0003), o tamanho do atributo (0002,0002, *Media Storage SOP Class UID*) sendo igual a 001A H = 26 D e por final o seu conteúdo (1.2.840.10008.5.1.4.1.1.2.). No lado direito a representação em ASCII e na esquerda a representação em Hexadecimal (FXEDIT, 2004).

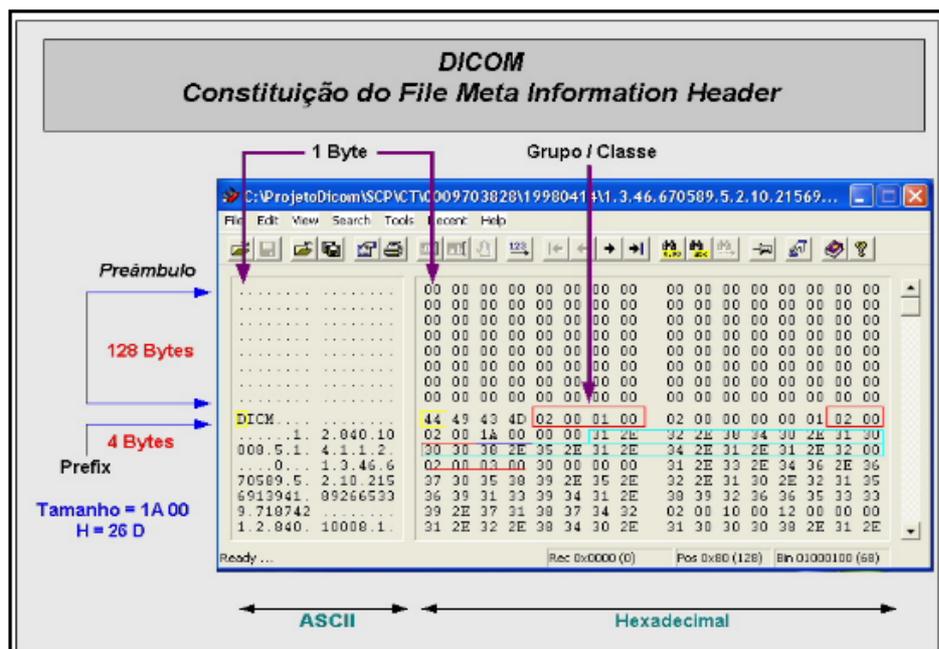


Figura 1: Formato do arquivo DICOM de uma imagem em hexadecimal.  
Fonte: FXEDIT, 2004.

### 3 PACS

Estas imagens podem ser facilmente integradas em ambientes que realizam diagnósticos com o uso de um PACS. Tais imagens podem ser armazenados digitalmente e posteriormente visualizadas de qualquer estação de trabalho, que pode ser configurada com diversos perfis de acesso, dentro de uma instituição. As estações podem estar dentro de um hospital, de um laboratório ou de uma clínica e podem estar projetadas para controle local do estado do paciente ou para consulta com especialistas em um local remoto. Da mesma forma, se o paciente recorrer a uma facilidade de atendimento terciária, as imagens e laudos deste paciente podem estar disponíveis na unidade original e podem ser acessados remotamente na unidade terciária obedecendo, para isto, protocolos de segurança.

### 4 Classe de Serviço e Classe SOP

A relação entre ambos - cliente e servidor - como parceiros é definida pela descrição da Classe de Serviço (*Service Class*). Esta descreve explicitamente os papéis ou regras (*Roles*) que estes parceiros desempenham.

Dependendo da Classe de Serviço em questão, é definido o contexto

(*Context*) dos serviços. No DICOM, os papéis são designados pela Classe de Serviço do Usuário (*Service Class User*) ou SCU e pela Classe de Serviço do Provedor (*Service Class Provider*) ou SCP.

Classe de Serviços é a descrição da informação e das operações. No padrão DICOM tais operações são combinadas com a definição da classe designada por Classe Par Objeto Serviço (*Service Object Pair Class*) ou abreviadamente Classe SOP (*SOP Class*).

A Sintaxe de Transferência é responsável pela Representação do Valor, Ordenação dos *Bytes* e o Método de Compressão, caso haja entre a Classe de Serviço do Usuário (Cliente) e a Classe de Serviço do Provedor (Provedor) envolvida durante a interação.

O mapa rodoviário demonstrado na figura 2 mostra os conceitos DICOM apresentados, anteriormente, no processo de codificação/decodificação da informação (DICOM, 2003).

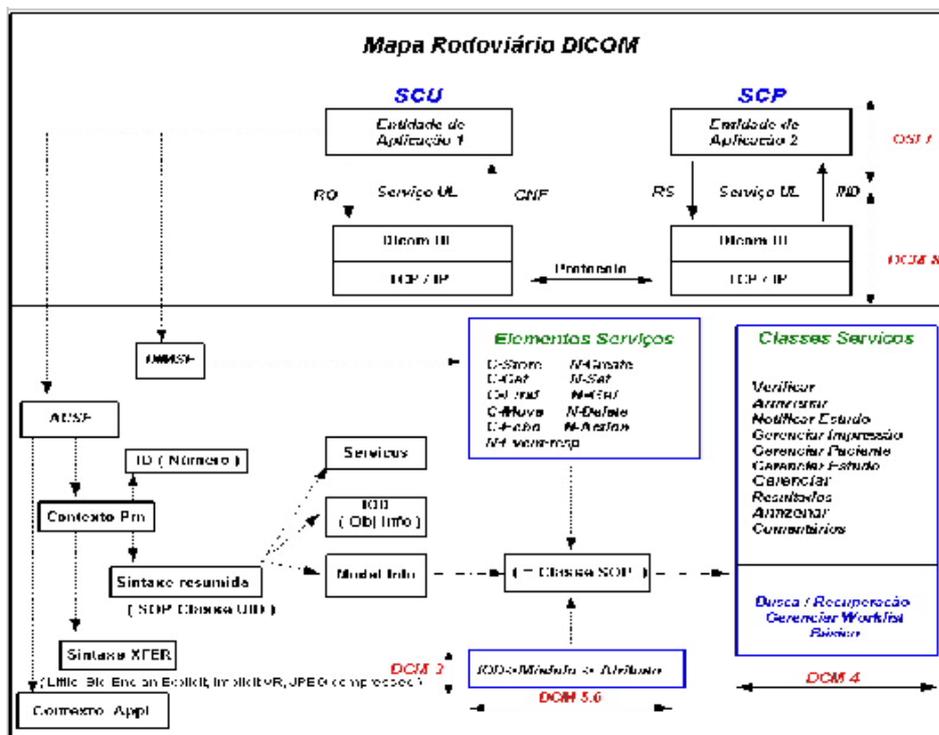


Figura 2 : Mapa rodoviário do padrão DICOM.  
 Fonte: DICOM, 2003.

No quadro 1 tem-se um resumo dos conceitos implementados na figura 2.

Entidades e Objetos	Information Object (IO)
Atributos que definem um objeto	Information Object Definition (IOD)
Métodos de um objeto	DICOM Message Service Elements (DIMSE)
Atributos de um objeto + Serviços	Service Object Pair (SOP = IOD + DIMSE)
Clientes de Serviços	Service Class User (SCU)
Servidor de Serviços	Service Class Provider (SCP)

Quadro 1: Conceitos DICOM.  
 Fonte: DICOM, 2003.

## 5 Conformidade de Comandos

A Conformidade de Comandos ou Sentenças (*Conformance Statement* – CS) nos especifica, em detalhes, a complexidade e a diversidade de tarefas que um produto deve ser capaz de realizar para estar dentro do padrão DICOM 3.0 (PRIOR, 1993).

## 6 Metodologia

Todo o *software* foi desenvolvido utilizando a linguagem de programação *Visual C++ 6.0 Enterprise Edition* da *Microsoft* para o ambiente *Windows*, juntamente com aplicações do *SQLAPI++* - para conexão com o banco de dados - e com a aplicação

do *Conquest DICOM Softwares* (CONQUEST, 2004), responsável pela comunicação e pela simulação de um equipamento gerador de imagens médicas (figura 3).

O protótipo se propõe a executar um algoritmo que fique monitorando a comunicação entre duas entidades. Quando percebe que uma imagem com informações agregadas está trafegando pela rede, o próprio algoritmo decodifica os grupos/elementos que estão transitando e automaticamente realiza uma inserção parametrizada dependendo dos atributos que compõe as entidades e armazena a(s) imagem(ns) juntamente com um ponteiro indicando a localização física do armazenamento da imagem.

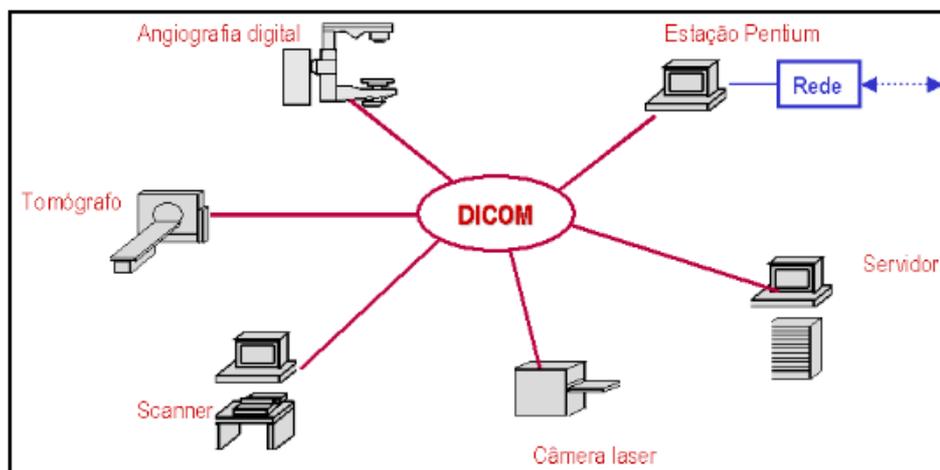


Figura 3 : Concepção da aplicação.  
Fonte: CONQUEST, 2004.

## 6.1 Modelagem

Sendo o padrão DICOM por definição orientada a objeto, observa-se o diagrama de classe na figura 4. Na maioria dos casos isso envolve a modelagem do vocabulário do sistema, a modelagem dos colaboradores ou a modelagem dos esquemas. São também importantes não só para a visualização, especificação e documen-

tação, assim como para a construção de sistemas executáveis por intermédio de engenharia de produção e reversa. Este modelo consiste de quatro entidades: paciente (*patient*), estudo (*study*), série (*serie*), e imagem (*image*). Por motivos de visibilidade, os métodos não estão representados o tornado semelhante ao Diagrama Entidade-Relacionamento (E/R) (CHEN, 1990).

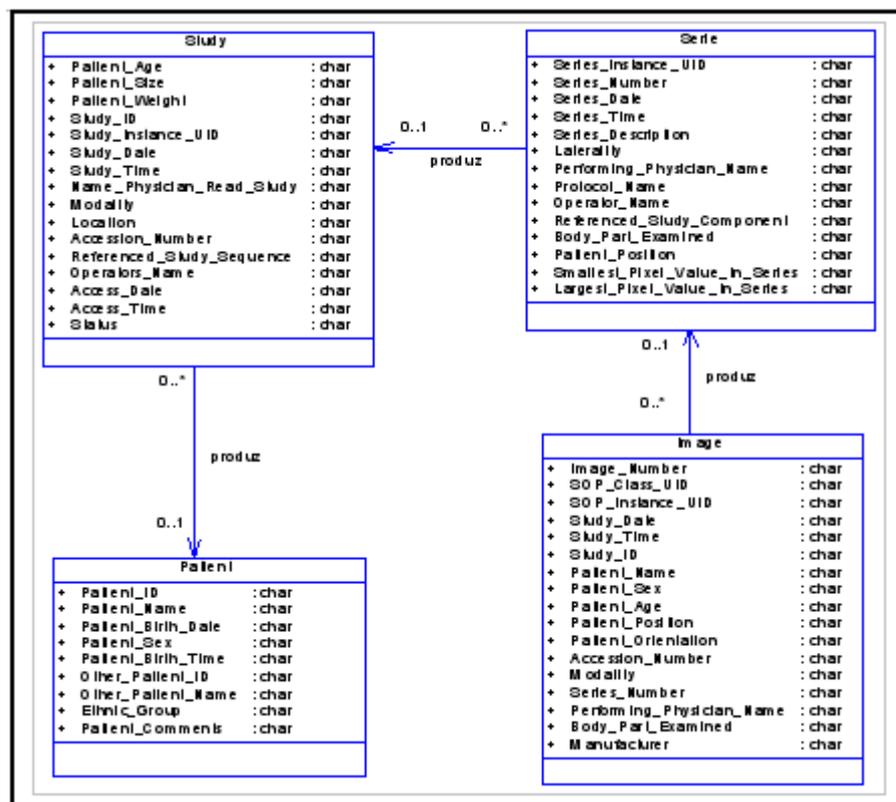


Figura 4 : Diagrama de classe.  
Fonte: CHEN, 1990.

## 6.2 Sistema gerenciador de banco de dados (SGDB)

É uma peça fundamental na infraestrutura de *software* de qualquer aplicação, seja ela de informática ou não. Um sistema gerenciador de banco de dados confiável deve apresentar uma série de funcionalidades, tais como: segurança de dados, consistência, disponibilidade, recuperação de falha, desempenho e controle de concorrência.

No entanto, os SGDBs comerciais existentes costumam cobrar muito caro por estes recursos, especialmente devido ao esquema de licenciamento empregado, que costuma levar em consideração o número de usuários simultâneos e/ou a configuração do servidor (DATE, 1992).

Seguindo o rastro de outras aplicações livres como navegadores, ferramentas de *e-mail*, servidores *web* e editores de texto, começam a aparecer opções competitivas de SGDB *Open Source* e gratuitos tais como *MySQL*, *PostgreSQL*, *Firebird*, *SapDB/MaxDB* e *BerkeleyDB*. Fala-se neste momento de algumas características dos SGDBs que foram utilizados na implementação da aplicação.

### 6.2.1 Postgresql

É o SGDB livre mais bem visto pela comunidade acadêmica. Isso se deve em parte ao fato de ter sido desenvolvido a partir do código do *Postgres*, na Universidade de Berkeley, em um projeto liberado pelo Professor Michael Stonebraker, até hoje um dos cientistas mais proeminentes na área de pesquisa em banco de dados.

Ele também foi líder do projeto que criou o *Ingres*, antepassado de banco de dados como *Sybase*, *SQL Server*, *Informix* e o próprio *Postgres*. O projeto final deste terminou no início dos anos 1990, porém dois alunos resolveram adicionar o suporte a linguagem *SQL* ao *Postgres*, dando origem ao *PostgreSQL*. (IEEE, 2007)

A coordenação do desenvolvimento e liberação de versões está a cargo do *PostgreSQL Global Development Group*, uma comunidade mantida por várias empresas ligadas ao banco de dados conectadas ao redor do mundo e diversas empresas oferecendo contratos de suporte ao mesmo. Na figura 5 é ilustrada a janela principal de uma ferramenta de administração de banco de dados chamada *pgAdmin III*. No lado esquerdo, pode-se observar o banco de dados criado “dicom\_postgresql”, inse-

rido neste banco de dados o esquema “dicomserver” juntamente com as tabelas *patient*, *study*, *serie* e *image*. No lado direito, observa-se a DDL (*Data Definition Language*) de criação da tabela *patient*.

### 6.2.2 Firebird

Este é um banco de dados livre que foi desenvolvido a partir do código fonte do *Borland InterBase 6.0*, liberado ao público em 2000. A origem do *Interbase* remonta a 1984, tornan-

do-se bastante popular entre os desenvolvedores que utilizam as linguagens da *Borland*, que pode ser distribuído juntamente com elas. Atualmente o código fonte do *Firebird* já foi bastante modificado, tendo sido portado de *C* para *C++* na versão 1.5, constituindo-se num produto que se distancia cada vez mais do *Interbase* original, tanto em desempenho quanto em funcionalidade.

A exemplo do *PostgreSQL*, o *Firebird* também é livre para todo o tipo de aplicação. Uma organização

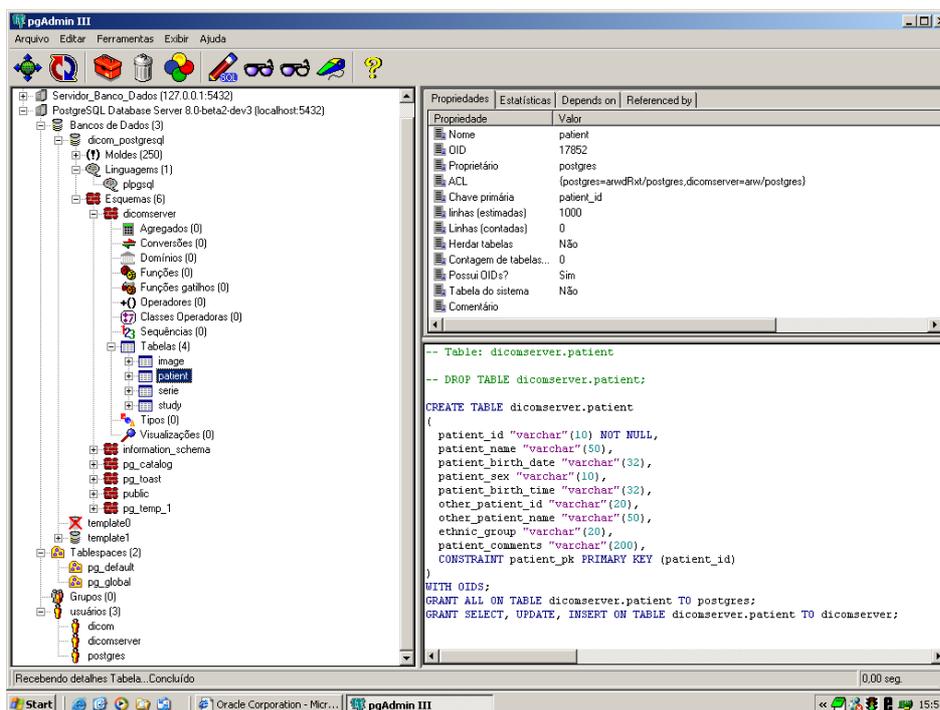


Figura 5 : Ferramenta de administração de banco de dados *PostgreSQL*.  
Fonte: PGADMIN III, 2007.

chamada *Firebird Foundation* foi criada com a função de arrecadar e distribuir fundos para os principais desenvolvedores do mesmo. A seguir é ilustrado uma ferramenta de administração de banco de dados chamada *IB Expert - The Most Expert for InterBase, Firebird and Yaffil*.

No lado esquerdo, pode-se observar o banco de dados criado “dicom\_firebird”, a tabela *patient*, *study*, *serie*, e *image* e a descrição dos atributos da tabela *patient*. No lado direito, observa-se um registro da tabela *patient* (figura 6).

## 6.2.3 Oracle

O *Oracle Server* é um sistema multiusuário de gerenciamento de banco de dados, ou seja, um produto de *software* especializado em gerenciar um único conjunto compartilhado de informações entre vários usuários concorrentes.

O *Oracle* suporta todos os principais sistemas operacionais, tanto para clientes quanto para servidores, dentre eles o *MS-DOS*, *NetWare*, *UnixWare*, *OS/2*, *Windows NT* e a maioria das variações do *UNIX*.

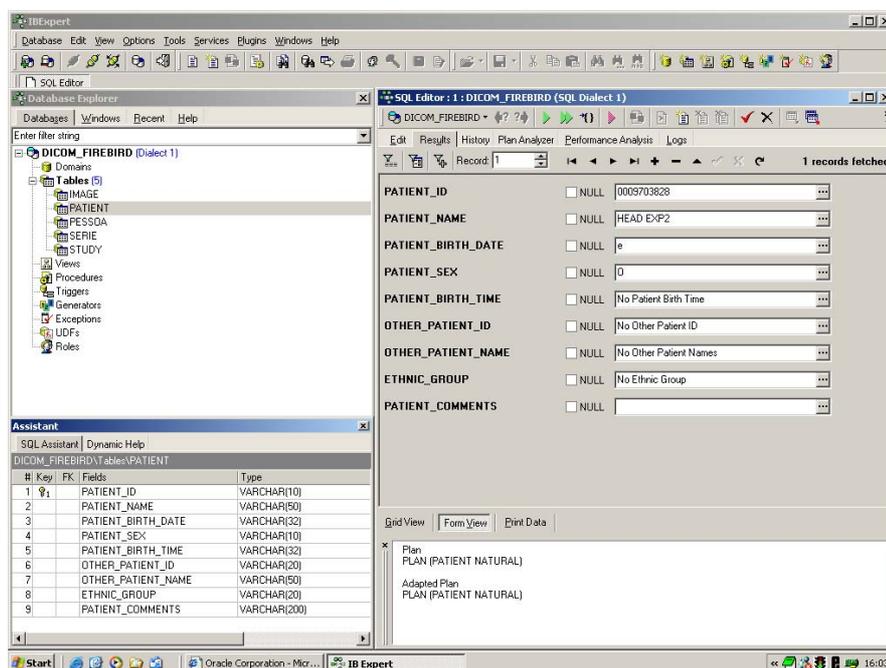


Figura 6 : Ferramenta de administração de banco de dados *Firebird*.  
Fonte: IB EXPERT, 2007.

O software de operação em rede da Oracle, o *SQL\*Net*, suporta todos os principais protocolos de comunicação de rede, tais como o TCP/IP, SPX/IPX, *Named Pipes* e DEC-Net.

As características do *Oracle Server* permitem armazenar e gerenciar informações com todas as vantagens de uma estrutura relacional mais os comandos PL/SQL, que é uma aplicação que provê a capacidade de armazenar e executar objetos do banco de dados, tais como *procedures* e *triggers*. Ele oferece aos usuários opções para a recuperação dos dados

baseando-se em técnicas de otimização, que inclui características de segurança que controlam como o banco de dados é acessado e utilizado.

Também oferece consistência e proteção dos dados através de mecanismos de bloqueios e processos de gravação *commit* em duas fases para os dados distribuídos através da rede. A seguir é ilustrado na figura 7 uma ferramenta de administração de banco de dados chamada *DBA Studio*. No lado esquerdo, pode-se observar o banco de dados criado “dicom\_oracle”, o esquema “dicomserver”, a tabela

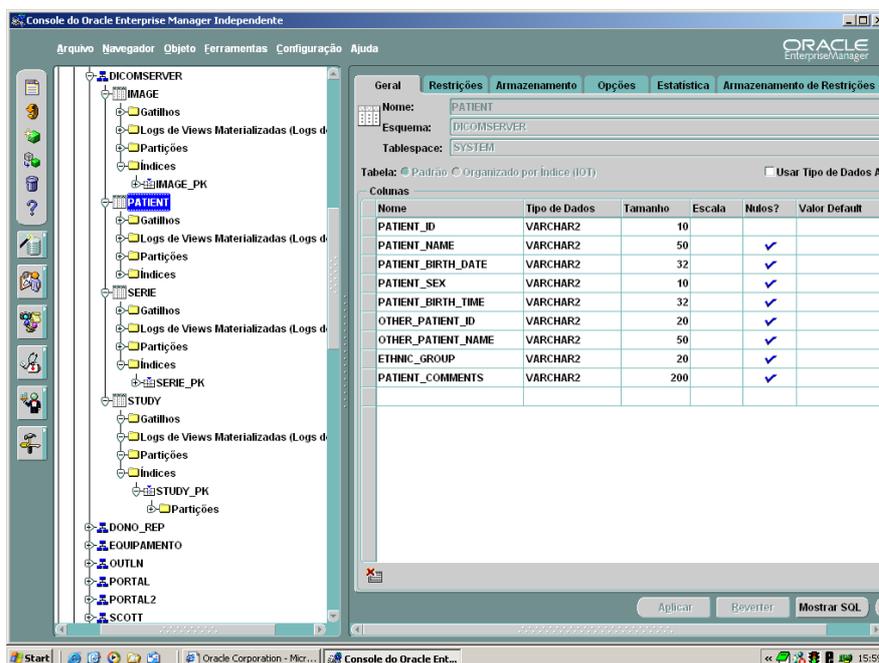


Figura 7 : Ferramenta de administração de banco de dados Oracle.  
Fonte: DBASTUDIO, 2007.

*patient, study, serie, e image*. No lado direito, observa-se a descrição dos atributos da tabela *patient*.

#### 6.2.4 Implementação

O Hospital Universitário de Brasília trabalha com o aparelho de Ultrasonografia “Philips HDI 3500 System” (figura 8), que gera imagens durante a realização de exames nos pacientes.



Figura 8 : Philips HDI 3500 System.  
Fonte: Elaborada pelo autor.

O objetivo é enviar imagens recolhidas do “HDI 3500” (figura 8) para a aplicação de modo que esta possa armazenar as imagens no disco rígido e inserir os dados em um sistema gerenciador de banco de dados previamente escolhido.

Após esta escolha, toda e qualquer imagem enviada do equipamento para a aplicação será armazenada no disco rígido e inserida no banco de dados respeitando as restrições de integridade. A porta previamente configurada é a 1010, que deverá estar contida no arquivo de configuração do aparelho médico.

Observa-se na figura 9 a coleta dos dados do paciente. Podem-se armazenar as imagens e dados provenientes dos respectivos exames na rede onde estão localizados o servidor de imagens e o servidor de banco dados.



Figura 9: Paciente no Hospital Universitário de Brasília.

Fonte: Elaborada pelo autor.

Observa-se na figura 10 a inicialização da aplicação que fica na expectativa da chegada dos exames, após a seleção de um banco de dados, *Firebird 1.5*.

Observa-se na figura 11 a criação da árvore de diretório.

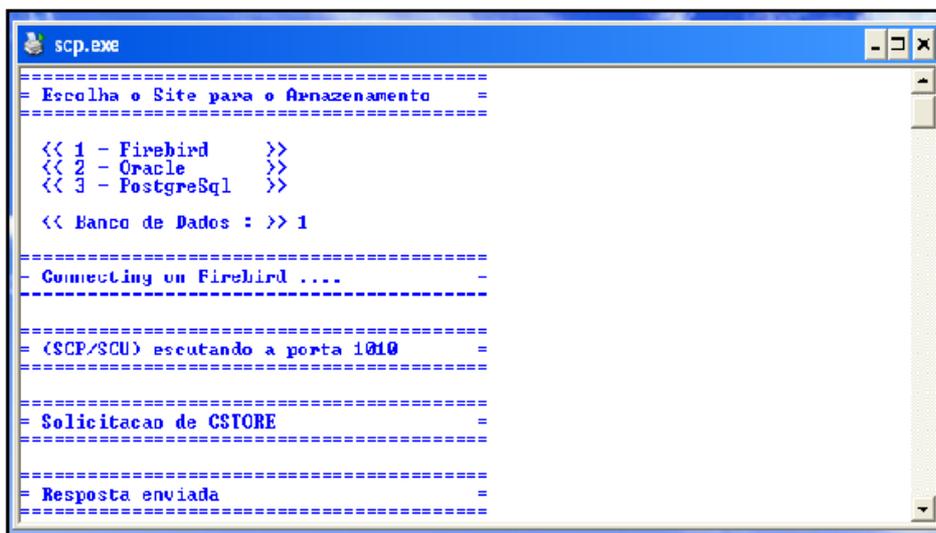


Figura 10 : Inicialização da aplicação.  
 Fonte: Elaborada pelo autor.

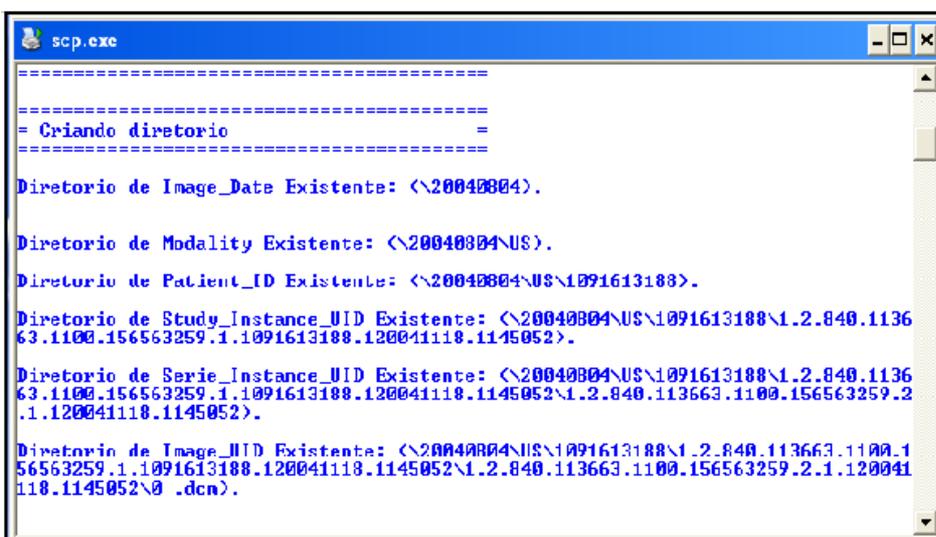


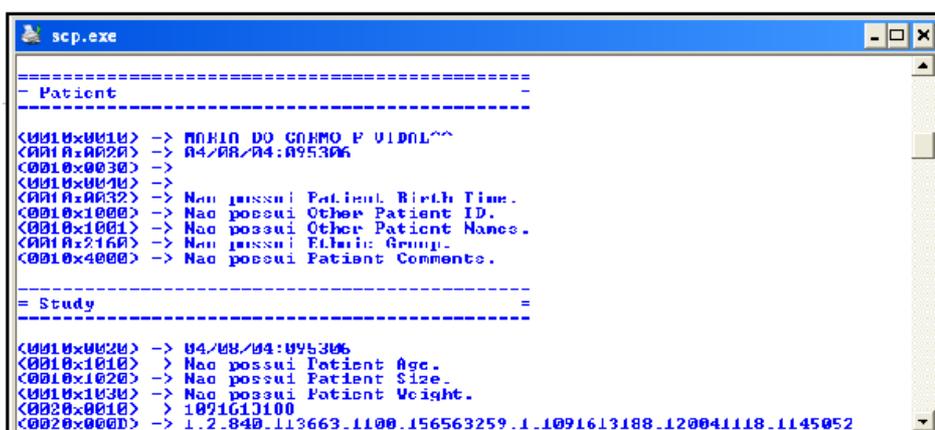
Figura 11 : Criação da árvore de diretório.  
 Fonte: Elaborada pelo autor.

Observa-se na figura 12 a coleta dos elementos de dados com os respectivos rótulos.

Observa-se na figura 13 a árvore de diretório construída com as respectivas imagens armazenadas no disco

rígido.

Observa-se na figura 14 os elementos de dados armazenados na tabela *patient*. E na figura 15 os endereços de armazenamento da imagem.



```
=====  
- Patient  
=====  
<0010x0010> -> nome do grupo P VIDAL^^  
<0010x0020> -> 04/08/04:095306  
<0010x0030> ->  
<0010x0040> ->  
<0010x0032> -> Não possui Patient Birth Time.  
<0010x1000> -> Não possui Other Patient ID.  
<0010x1001> -> Não possui Other Patient Names.  
<0010x2160> -> Não possui Ethnic Group.  
<0010x4000> -> Não possui Patient Comments.  
  
=====  
= Study  
=====  
<0010x0020> -> 04/08/04:095306  
<0010x1010> -> Não possui Patient Age.  
<0010x1020> -> Não possui Patient Size.  
<0010x1030> -> Não possui Patient Weight.  
<0020x0010> -> 1071613100  
<0020x000D> -> 1.2-840.113663.1100.156563259.1.1091613188.120041118.1145052
```

Figura 12 : Atributos da tabela *patient* e *study*.  
Fonte: Elaborada pelo autor.

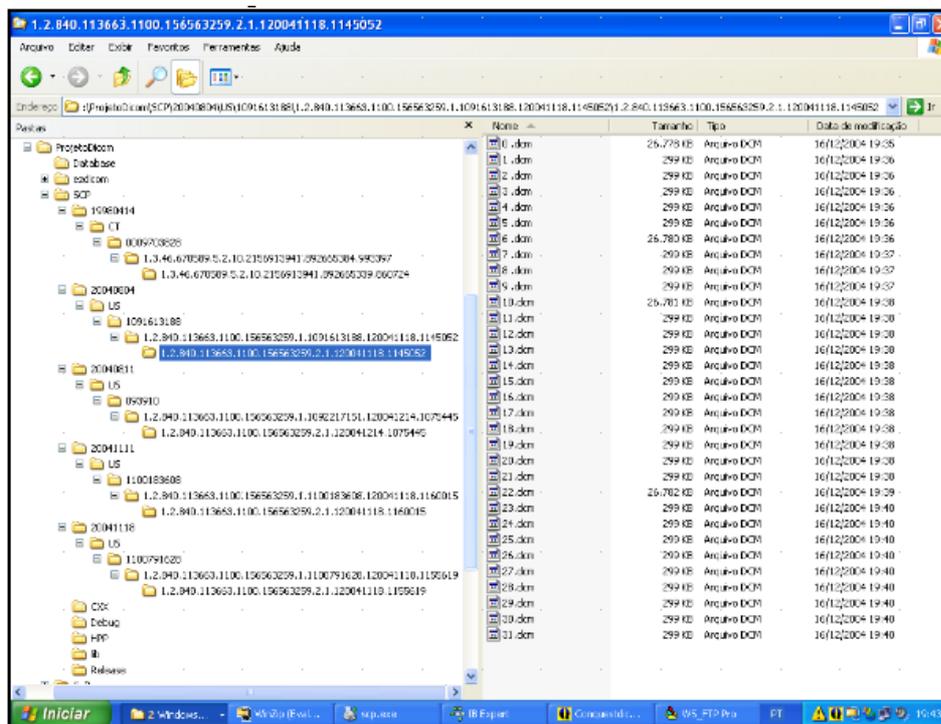


Figura 13 : Criação da árvore de diretórios.  
Fonte: Elaborada pelo autor.

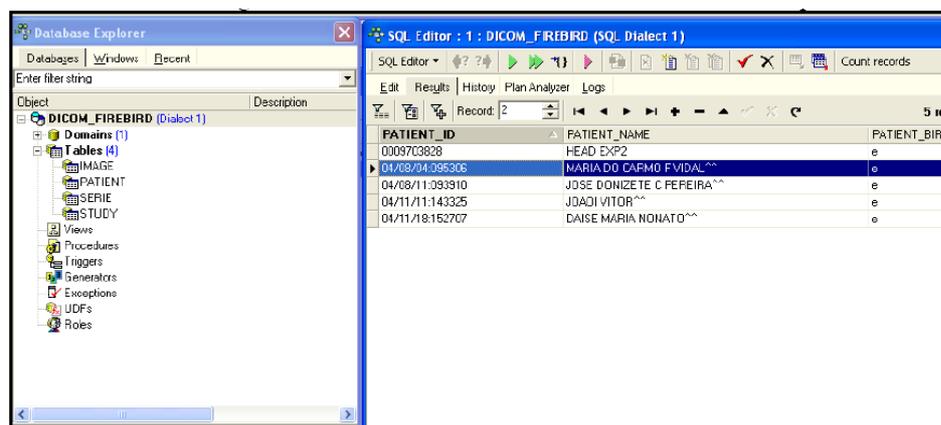


Figura 14 : Tabela *patient*.  
Fonte: Elaborada pelo autor.



Figura 15 : Tabela *image* endereço “2000.dcm”.  
Fonte: Elaborada pelo autor.

Observa-se na figura 16 uma das imagens capturadas armazenada no disco rígido (EZDICOM, 2004).

Observa-se na figura 17 um conjunto de imagens capturadas armaze-

nado no disco rígido onde é possível o estudo e a comparação entre elas pela comunidade médica (EZDICOM, 2004).

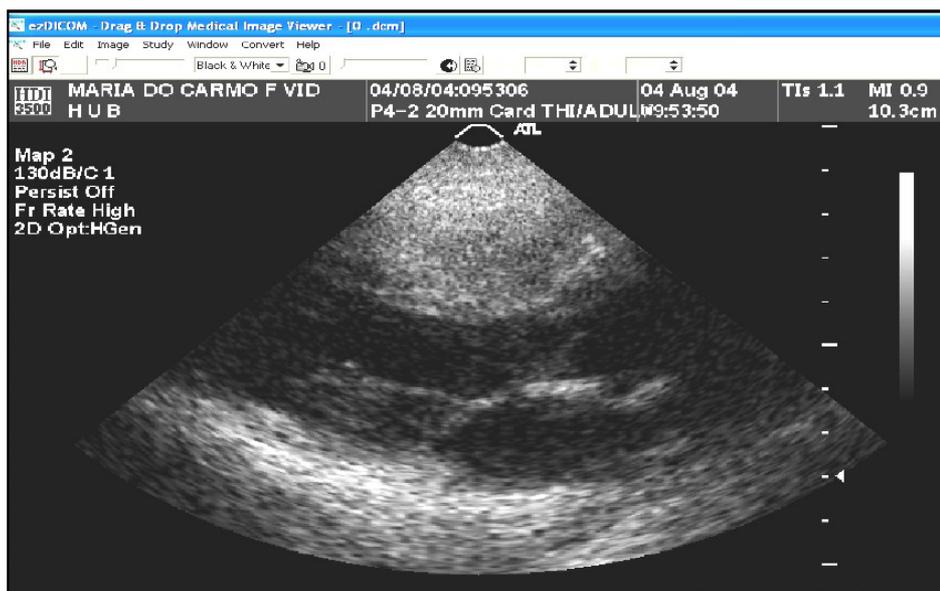


Figura 16 : Visualização da imagem enviada.  
Fonte: EZDICOM, 2004.

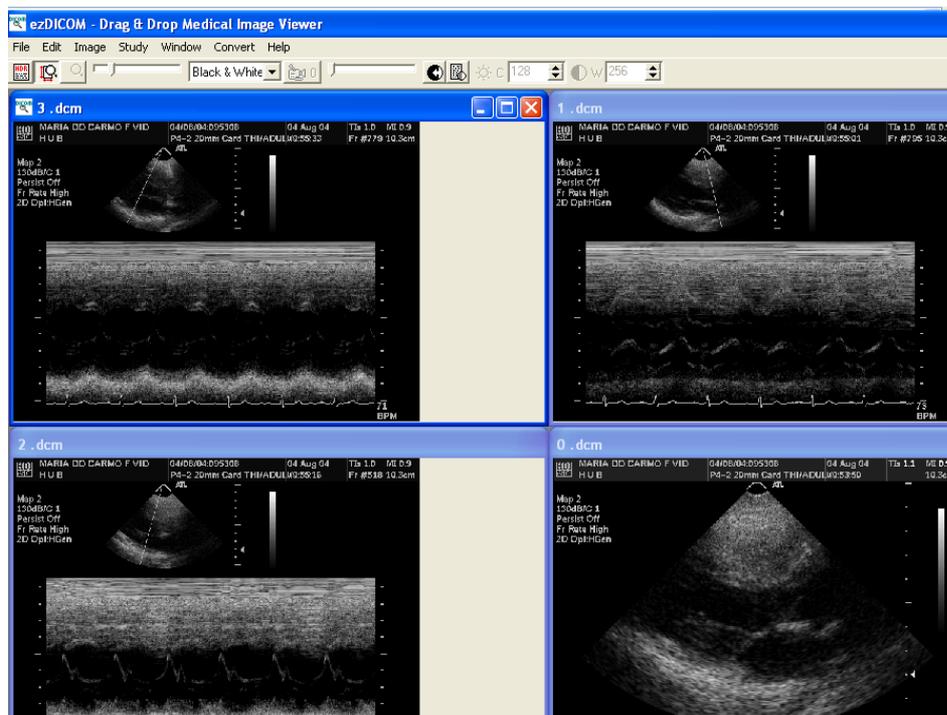


Figura 17 : Visualização das imagens enviadas.  
 Fonte: EZDICOM, 2004.

## 7 Conclusão

Este projeto foi proposto pela necessidade de constatar um dos mais importantes problemas decorridos do expressivo progresso da radiologia no tratamento radioterápico, que é o armazenamento de imagens e disponibilidade das informações de maneira rápida e consistente.

A abordagem do problema objetivou a utilização de conceitos fundamentais do padrão de comunicação

e arquivamento, conhecido mundialmente como DICOM e que é um dos principais pilares para a implementação de um PACS e de um prontuário eletrônico. O trabalho propôs uma aplicação robusta, eficiente e de baixo custo computacional com o objetivo de automatizar o processo de comunicação e arquivamento de imagens e informações agregadas de forma transparente e consistente. Esta proposta foi executada após uma revisão bibliográfica sobre as partes que compõem

o protocolo DICOM, códigos fontes abertos provenientes da *Internet*, que poderiam nos auxiliar na implementação e no estudo de sistemas gerenciadores de banco de dados.

Em seguida, o sistema foi implementado e testado no Hospital Universitário de Brasília durante a realização de exames médicos utilizando o equipamento de ultra-sonografia “Philips HDI 3500 System”, responsável pela comunicação e transmissão das imagens dos exames para a aplicação desenvolvida neste trabalho.

## Referências

ALMEIDA, A.B. Usando o computador para o processamento de imagens médicas. **Informática médica**. Campinas, v.1, n. 6, nov/dez 1998. Disponível em: <<http://www.informaticamedica.org.br/informaticamedica/n0106/index.html>>. Acesso em: 28 nov. 2007.

BAXTER, B.S.; HITCHNER, L.E.; MAGUIRE Jr, G.Q. A standard format for digital image exchange. **American Association of Physicists in Medicine**, New York, n. 10, mar 1982. Disponível em: <[http://www.aapm.org/pubs/reports/rpt\\_10.pdf](http://www.aapm.org/pubs/reports/rpt_10.pdf)>. Acesso em: 30

nov. 2007.

BOOCH, G. **Object-Oriented Analysis and Design with Applications**. 2 ed. Reading: Addison Wesley Publishing Company, 1993.

CHEN, P. **Modelagem de Dados**. São Paulo: McGraw-Hill, Makron, 1990.

CONQUEST DICOM *software*. Versão 1.4.13. Disponível em: <<http://www.xs4all.nl/~ingenium/dicom.html>>. Acesso em: 04 jun. 2004.

DATE, C.J. **Introdução a Sistemas de Banco de Dados**. Rio de Janeiro: Campus, 1992.

DBASTUDIO. *Oracle Corporation*. Disponível em: <<http://www.oracle.com>>. Acesso em: 29 nov. 2007.

DICOM - Digital Imaging and Communication in Medicine. NEMA. Disponível em: <<http://medical.nema.org>>. Acesso em: 04 dez. 2003.

EZDICOM *Company*. Versão 1, *release 19*. *Open Source*

*Technology Group*. Disponível em: <<http://www.psychology.nottingham.ac.uk/staff/cr1/ezdicom.html>>. Acesso em: 10 jan. 2004.

FXEDIT Company. Versão 3.5. Disponível em: <<http://superdownloads.ubbi.com.br/download/i9811.html>>. Acesso em: 10 jan. 2004.

GONZALEZ, R. C.; WOODS, R. E. **Processamento de Imagem Digital**. São Paulo: Edgard Blücher, 2000.

IB EXPERT. HK *Software*. Disponível em: <<http://info.abril.com.br/download/4047.shtml>>. Acesso em: 29 nov. 2007.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS - IEEE. **Michael Stonebraker**. Piscataway, 2007. Disponível em: <<http://www.ieee.org/portal/pages/about/awards/bios/2005vonneumann.html>>. Acesso em: 03 dez. 2007.

PGADMIN III. Versão 1.5. Disponível em: <[http://www.postgresql.org.br/Ferramentas\\_para\\_](http://www.postgresql.org.br/Ferramentas_para_)

[o\\_PostgreSQL](http://www.postgresql.org.br/Ferramentas_para_o_PostgreSQL)>. Acesso em: 29 nov. 2007.

PRIOR, F.W. Specifying DICOM compliance for modality interfaces. **RadioGraphics**, Oak Brook, v. 13, iss. 6, p.1381-1388, nov 1993. Disponível em: <<http://radiographics.rsna.org/cgi/reprint/13/6/1381.pdf>>. Acesso em: 30 nov. 2007.

SQLAPI++ . Versão 3.7.11 . Disponível em: <<http://www.sqlapi.com/>>. Acesso em: 10 mar. 2004.

UNIVERSIDADE FEDERAL DE SÃO PAULO. Departamento de Informática em Saúde. **DICOM – Comunicação de imagem digital em medicina**. São Paulo, 2004. Disponível em: <<http://www.virtual.epm.br/material/tis/curr-med/temas/med5/med5t21999/dicom/dicom2.htm>>. Acesso em: 04 jun. 2004.