

UNIVERSIDADE DO EXTREMO SUL CARATINENSE – UNESC

CURSO DE CIÊNCIA DA COMPUTAÇÃO

RICARDO DO AMARAL TOLFO

**APLICAÇÃO MÓVEL PARA ACESSO A UM SISTEMA DE REGISTRO
ELETRÔNICO EM SAÚDE**

CRICIÚMA, JULHO DE 2009

RICARDO DO AMARAL TOLFO

**APLICAÇÃO MÓVEL PARA ACESSO A UM SISTEMA DE REGISTRO
ELETRÔNICO EM SAÚDE**

Trabalho de Conclusão de Curso apresentado
para obtenção do Grau de Bacharel em Ciência
da Computação da Universidade do Extremo
Sul Catarinense.


Orientador: Prof. MEng. Evânio Ramos Nicoleit

CRICIÚMA, JULHO DE 2009

RICARDO DO AMARAL TOLFO


Aplicação Móvel para Acesso a um Sistema de Registro Eletrônico em Saúde

Submetido ao corpo docente do Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense como um dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.


Prof. MSc. Rogério Antônio Casagrande
Coordenador Adjunto do Curso de Ciência da Computação

Banca Examinadora:


Prof. MEng. Evânio Ramos Nicoleit (UNESC)
Orientador


Prof. MSc. Paulo João Martins (UNESC)
Banca Examinadora


Prof. Fabrício Giordani (UNESC)
Banca Examinadora

*Dedico este trabalho a minha mãe e minha irmã
que sempre me apoiaram para que eu
conseguisse realizar com sucesso todos os
desafios.*

AGRADECIMENTOS

Primeiramente, agradeço a Deus por ter me dado saúde, paz e força para que eu superasse todos os meus obstáculos.

Ao meu orientador Evânio e aos colegas do Kiron que me ajudaram nas horas de dificuldade, dedicando a atenção necessária ao longo desse projeto.

Aos meus colegas de curso, aos quais compartilhei bons momentos durante esses anos.

Aos meus amigos, que nos momentos difíceis, me deram todo o apoio moral, entenderam meu afastamento e por várias vezes que eu ouvi “vai te coça!”, fazendo com que eu não desistisse.

À minha mãe e à minha irmã, as quais devo minha vida, estiveram sempre ao meu lado em todos os momentos, à minha mãe que me transformou em uma pessoa responsável e dedicada.

E por último, ao meu finado pai, que me transformou em um homem, transmitiu toda sua experiência de vida, sabedoria e caráter, no qual sempre me espelhei para me tornar uma pessoa melhor.

*“A esperança não é um sonho, mas uma
maneira de traduzir os sonhos em realidade.”*
Cardeal Suenens

RESUMO

Um Sistema de Registro Eletrônico em Saúde (S-RES) é utilizado no gerenciamento de informações sobre os pacientes ou sobre as internações em uma Unidade de Terapia Intensiva (UTI). Atualmente, profissionais de saúde requerem acesso às informações, independentemente de lugar ou horário. Para prover mobilidade de consulta ao S-RES, foi desenvolvida uma aplicação para acesso às informações contidas no sistema por meio de um cliente utilizando um Dispositivo Móvel (DM), executando nele uma aplicação com acesso a um banco de dados, por meio da rede wireless. As informações são enviadas de forma segura, utilizando métodos de criptografia e certificação digital. Neste projeto são abordados também as tecnologias requeridas para o desenvolvimento de uma aplicação móvel para acesso ao S-RES e apresentada uma interface para o mesmo.

Palavras-Chave: Sistema de Registro Eletrônico de Saúde; Mobilidade; Dispositivo Móvel.

ABSTRACT

An Electronic Health Record System (EHR) is used in the management of information on the patients or on the admissions in an Intensive Care Unit (ICU). Currently, health professionals need access to information, regardless of place or time. To provide mobility to consulting the EHR, an application was developed for access to information in the system by a client using a Mobile Device (MD), running an application with access to a database through the wireless network. The information is sent in a secure way, using methods of encryption and digital certification. In this project also addressed the technologies required to develop a mobile application to access the EHR and presented an interface for it.

Keywords: *Electronic Health Record System; Mobility; Mobile Device.*

LISTA DE ILUSTRAÇÕES

Figura 1. Modelo RMOP.....	27
Figura 2. Demonstração de envio de mensagem utilizando o padrão TISS.....	33
Figura 3. Palm m130.....	41
Figura 4a. Nokia E71.....	42
Figura 4b. Nokia E61.....	42
Figura 5. HP Compaq Tablet PC tc1100.....	43
Figura 6. Plataformas Java.....	46
Figura 7. Arquitetura Servlet.....	49
Figura 8. Arquitetura JSP	50
Figura 9. Fluxograma da comunicação do S-RES com o Banco de Dados.....	68
Figura 10. Diagrama de Classe Banco de Dados UTInfo 2.0.....	69
Figura 11. Diagrama de Caso de Uso das funcionalidades do mUTInfo 2.0.....	70
Figura 12. Tela de Login mUTInfo 2.0.....	71
Figura 13. Tela de <i>Status</i>	72
Figura 14. Tela de alerta de erro.....	72
Figura 15. Tela Procura Pacientes.....	73
Figura 16. Tela Carrega Pacientes.....	74
Figura 17. Carregar Dados Gerais.....	74
Figura 18. Tela de Dados Gerais do Paciente.....	75
Figura 19. Tela de Dados iniciais.....	76
Figura 20. Tela de Diagnósticos de Entrada.....	76
Figura 21. Tela de Condições Gerais.....	76
Figura 22. Tela de Evolução Médica.....	76

Figura 23. Classe HomeDir.....78

LISTA DE QUADROS

Quadro 1. Vantagens do uso de um S-RES.....	23
Quadro 2. Desvantagens do uso de um S-RES.....	24

LISTA DE SIGLAS

API	Application Programming Interfaces
BD	Banco de Dados
CDC	Connected Device Configuration
CDMA	Code Division Multiple Access
CGI	Common Gateway Interface
CID	Classificação Internacional de Doenças
CLDC	Connected Limited Device Configuration
CM	Computação Móvel
CPU	Central Processing Unit
DAO	Data Access Object
DM	Dispositivos Móveis
EDGE	Enhanced Data rates for GSM Evolution
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HL7	Health Level Seven
HTML	HyperText Markup Language
HTTP	HiperText Transfer Protocol
HTTPS	HiperText Transfer Protocol Security
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IoC	The Inversion of Control
IrDA	Infrared Data Association
JCP	<i>Java Community Process</i>

J2EE	Java Enterprise Edition
J2ME	Java Micro Edition
J2SE	Java Standard Edition
JDBC	Java Data Base Connective
JDO	Java Data Objects
JSP	Java Server Pages
JVM	Java Virtual Machine
LOINC	Logical Observation Identifiers Names and Codes
MIDP	Mobile Information Device Profile
ODBC	Open Database Connectivity
ORM	Object / Relational Mapping
S-RES	Sistema de Registro Eletrônico de Saúde
PDA	Personal Digital Assistant
POMR	Problem Oriented Medical Records
SDO	Standards Development Organization
SGBD	Sistemas Gerenciadores de Bancos de Dados
SMS	Short Message Service
SNOMED	Systematized Nomenclature of Medicine
SQL	Structured Query Language
TDMA	Time Divison Multiple Access
TI	Tecnologia de Informação
TISS	Troca de Informações em Saúde Suplementar
UNESC	Universidade do Extremo Sul Catarinense
UTI	Unidade de Terapia Intensiva
WAP	Wireless Application Protocol

W-CDMA	Wideband Code Division Multiple Access
WEP	Wireless Equivalent Privacy
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WLL	Wireless Local Loop
WMAN	Wireless Metropolitan Area Network
WML	Wireless Markup Language
WPAN	Wireless Personal Area Network
WWAN	Wireless Wide Area Network
XML	eXtensible Markup Language

SUMÁRIO

1	INTRODUÇÃO.....	17
1.1	OBJETIVO GERAL.....	18
1.2	OBJETIVOS ESPECÍFICOS	19
1.3	JUSTIFICATIVA	19
1.4	ESTRUTURA DO TRABALHO	20
2	SISTEMA DE REGISTRO ELETRÔNICO DE SAÚDE – S-RES	22
2.1	PRONTUÁRIO DO PACIENTE	22
2.2	S-RES - DEFINIÇÃO	23
2.3	S-RES: VANTAGENS E DESVANTAGENS	24
2.5	PADRÕES DE REGISTRO E TRANSMISSÃO DE DADOS EM SAÚDE.....	28
2.5.1	Padrão <i>OpenEHR</i>	30
2.5.2	Padrão TISS	32
2.5.3	Padrão HL7 (<i>Health Level Seven</i>)	34
2.5.4	Classificação Internacional de Doenças (CID)	34
2.5.5	XML para Comunicação	35
2.6	A ÉTICA NA UTILIZAÇÃO DO S-RES E ASPECTOS LEGAIS	35
3	COMPUTAÇÃO MÓVEL	37
3.1	DISPOSITIVOS MÓVEIS	37
3.1.1	TELEFONES MÓVEIS CELULARES	38
3.1.2	PDA	40
3.1.3	SMARTPHONES	41
3.1.4	TABLET PC	42
4	TECNOLOGIAS ASSOCIADAS AO DESENVOLVIMENTO DE APLICAÇÕES E USO DE DISPOSITIVOS MÓVEIS.....	44
4.1	TECNOLOGIA JAVA	44
4.1.1	Java Enterprise Edition	46
4.1.1.1	JDBC	47
4.1.1.2	<i>Servlets</i>	48
4.1.1.3	JSP	49
4.1.2	Java Standard Edition	50
4.1.3	Java Micro Edition	51
4.1.3.1	Configurações	52
4.1.3.1.1	Connected Device Configuration (CDC)	53
4.1.3.1.2	Connected Limited Device Configuration (CLDC)	53
4.1.3.1	Perfil	54
4.1.3.1.1	Mobile Information Device Profile (MIDP)	55
4.2	ACESSO AO BANCO DE DADOS UTILIZANDO JAVA	56
4.3	REDES DE COMPUTADORES	57
4.3.1	Wi-Fi	58
4.3.1.1	Endereçamento MAC	59
4.3.1.2	<i>Wired Equivalent Privacy (WEP)</i>	59
4.3.1.3	<i>Wi-fi Protected Access (WPA)</i>	60
4.3.1.4	Certificação Digital.....	60

4.4	<i>FRAMEWORKS</i>	61
4.4.1	<i>Hibernate</i>	61
4.4.2	<i>SpringMVC</i>	62
5	TRABALHOS CORRELATOS	63
6	TRABALHO DESENVOLVIDO	65
6.1	COMUNICAÇÃO PRONTUÁRIO - DM	65
6.2	O REGISTRO ELETRÔNICO DE PACIENTE - mUTInfo 2.0.....	68
6.3	RESULTADOS OBTIDOS	77
6.4	PROBLEMAS ENCONTRADOS	78
	CONCLUSÃO	79
	REFERÊNCIAS	80

1 INTRODUÇÃO

A Mobilidade cada vez mais faz parte do cotidiano da prática da Medicina. O profissional de saúde atual requer acesso a informações de um prontuário independentemente de restrições de tempo e espaço. As tecnologias móveis provêm mobilidade, por meio da interação com os sistemas, atendendo as necessidades de seus pacientes, de forma mais ágil, por meio do acesso às suas informações. Os profissionais podem acessar os documentos de seus pacientes, a distância, por meio de Dispositivos Móveis (DM) ligados como telefones celulares, PDAs, e Notebooks conectados à Internet.

Diariamente, pessoas que prestam serviços de cuidado e assistência à saúde se empenham em manter registros atualizados do estado de saúde, enfermidades e cuidados prestados a cada paciente (DORILEO et al, 2006). Contudo, informações resultantes de um atendimento médico, ou de enfermagem, podem ser perdidas, ou não registradas formalmente, ou ainda gerar esforços demasiados para o armazenamento, caso o médico esteja longe de sua estação de trabalho (SALOMÃO; SIGULEM, 2002). A utilização de um dispositivo móvel com acesso a um sistema de informações pode reduzir a perda de tais informações.

O Sistema de Registro Eletrônico de Saúde (S-RES) é um dos temas de pesquisa que tem despertado grande interesse na área da Informática Médica. O desenvolvimento do S-RES está associado a uma das linhas de pesquisa do Grupo de Pesquisa Informática Médica e Telemedicina - Kiron¹.

Tem sido desenvolvido, nos últimos anos, um sistema de registro eletrônico para acompanhamento médico dos pacientes em uma UTI. O sistema contém informações sobre a

¹ Grupo de Pesquisa Informática Médica e Telemedicina da Unesc - Projeto Kiron. É constituído por pesquisadores (professores e estudantes) que desenvolvem trabalhos de pesquisa em Informática Médica e Telemedicina.

internação de um paciente, como tratamento, medicamentos, tipo de doença, data de internação, evolução médica, entre outros. O S-RES foi desenvolvido com o intuito de prover aos médicos e enfermeiros do hospital um registro sistemático de fatos e acontecimentos clínicos sobre os pacientes.

Há profissionais de saúde que realizam seus atendimentos em um lugar fixo que possibilita o uso de um computador tipo *desktop* e um sistema de S-RES completo. Mas há também os profissionais de saúde móveis, que estão em constante movimento em seus ambientes e necessitam cada vez mais de ferramentas que acompanhem essa rotina (CRISPIM JUNIOR; FERNANDES, 2006). O desafio do S-RES atual é disponibilizar as informações que este possui para o profissional de saúde, em qualquer lugar que ele necessite e a qualquer momento. O sistema atende os padrões existentes de S-RES e de transmissão de dados por tecnologia sem fio.

Este trabalho descreve o desenvolvimento de uma aplicação para o acesso a informações do prontuário entre um servidor principal e um cliente utilizando um Dispositivo Móvel, executando nele uma aplicação com acesso a um banco de dados remoto, por meio de uma rede *wireless*. Serão abordadas, neste trabalho, as tecnologias requeridas para o desenvolvimento de uma aplicação móvel para acesso ao registro eletrônico para acompanhamento de pacientes, tecnologias para armazenamento e comunicação de dados do servidor principal para o dispositivo móvel e características de projeto de aplicações para DM.

1.1 OBJETIVO GERAL

Desenvolver um protótipo de uma aplicação para o acesso a informações do prontuário entre um servidor remoto e um cliente utilizando um DM por meio de uma rede *wireless*.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos desse projeto são descritos abaixo:

- a) analisar aspectos que envolvem o acesso ao sistema S-RES por meio de Dispositivos Móveis;
- b) estudar as tecnologias envolvidas para o desenvolvimento do sistema de S-RES Móvel;
- c) estudar as formas de acesso à base de dados do sistema de S-RES por meio dos DM;
- d) acesso a informações do Registro Eletrônico, no dispositivo móvel, por meio de dados exportados no formato XML.

1.3 JUSTIFICATIVA

Tem havido recentemente um grande avanço tecnológico no desenvolvimento de dispositivos para comunicação *wireless* a fim de disponibilizar cada vez mais mobilidade e interatividade aos usuários dessas tecnologias. Neste aspecto, os Dispositivos Móveis (DM) podem ser utilizados como plataforma para aplicações na área da saúde. Eles são tipicamente leves, versáteis, baratos, e podem ser utilizados virtualmente acessando informações à distância.

Profissionais de saúde que trabalham em constante movimento necessitam de acesso estável para acompanhamento dos pacientes. Esses perfis de profissionais na área da saúde ilustram situações em que o profissional tanto apresenta trânsito local, ou seja, desloca-se frequentemente dentro da instituição em que trabalha, como, por exemplo, em ambulatórios, emergências e pronto-socorros, quanto em situações onde os profissionais

atuam fora dos limites da instituição e atendem e/ou acompanham pacientes geograficamente distantes.

A Computação Móvel (CM) é aplicável ao contexto da Informática Médica e da Telemedicina. A CM melhora o processo de documentação de informações e codificação, provêm suporte a decisão e ao diagnóstico, e reduz o tempo total despendido em um atendimento devido à facilidade de acesso à base de informações do sistema (CRISPIM JUNIOR; FERNANDES, 2006).

Portanto, as aplicações da CM se mostram adequadas à Informática Médica e à Telemedicina, pois tendem a prover maior facilidade de acesso a informações.

1.4 ESTRUTURA DO TRABALHO

Essa pesquisa é composta de 6 capítulos. O primeiro capítulo relata uma introdução à pesquisa que foi desenvolvida, onde também estão descritos objetivos e a justificativa para a realização do trabalho.

No Capítulo 2 é feita uma descrição sobre o Sistema de Registro Eletrônico de Saúde (S-RES), suas vantagens e dificuldades de implantação, histórico do registro e padronização.

Os Dispositivos Móveis (DM) serão abordados no Capítulo 3, onde os tipos de DM, características e uma descrição sobre computação móvel serão abordados.

A tecnologia Java, suas características, funcionalidades, utilização, vantagens e atualidades sobre a mesma, são descritas no Capítulo 4.

No Capítulo 5 são mostrados e discutidos os trabalhos correlatos a essa pesquisa.

O trabalho desenvolvido, a descrição do S-RES analisado, resultados obtidos, ética para os profissionais de saúde e problemas encontrados no desenvolvimento do trabalho, estão apresentados no Capítulo 6.

Por fim, têm-se a Conclusão desse trabalho, onde são abordadas as considerações finais e também são sugeridos alguns temas para trabalhos futuros.

2 SISTEMA DE REGISTRO ELETRÔNICO DE SAÚDE – S-RES

A utilização das Tecnologias de Informação e Comunicação (TIC) na prática médica atual é uma realidade que facilita a organização, o armazenamento e o processamento das informações dos pacientes, apóia os profissionais na elaboração de diagnósticos precisos e auxilia na orientação terapêutica adequada.

2.1 PRONTUÁRIO DO PACIENTE

O prontuário do paciente (ou prontuário médico) é um elemento importante no atendimento à saúde dos indivíduos. O prontuário deve reunir informações que garantam a continuidade dos tratamentos aos pacientes (MARIN; MASSAD; AZEVEDO NETO, 2003).

O prontuário do paciente consiste em um conjunto de documentos, padronizados e ordenados, proveniente de diversas fontes, destinados ao registro dos cuidados prestados aos pacientes. O prontuário do paciente possui importância para paciente, instituição, profissionais de saúde e conseqüentemente para a sociedade (THOFEHRN; LIMA, 2007).

O Conselho Federal de Medicina (CFM), no Artigo 1º da Resolução de nº 1.638/2002, define o prontuário médico como o documento único constituído de um conjunto de informações, sinais e imagens registradas, geradas a partir de fatos, acontecimentos e situações sobre a saúde do paciente e a assistência a ele prestada, de caráter legal, sigiloso e científico, que possibilita a comunicação entre membros da equipe multiprofissional e a continuidade da assistência prestada ao indivíduo (CFM, 2009).

2.2 S-RES - DEFINIÇÃO

Em 1996, Peter Waegemann (1996), então presidente do *Medical Record Institute* (MRI), divide o S-RES em cinco níveis evolutivos, por ordem de complexidade:

- a) Registro Médico Automatizado (*Automated medical record*): Neste nível de sistema, a informação é armazenada em computadores pessoais e não está em acordo com os requisitos legais e, portanto, o prontuário em papel é mantido em conjunto. Desta forma, papel e registro eletrônico coexistem;
- b) Registro Médico Computadorizado (*Computerized medical record*): neste nível, médicos e toda a equipe coletam a informação no papel e a imagem dos documentos resultantes é armazenada de forma digitalizada no sistema computacional. Em geral, esse tipo de sistema é departamentalizado, com pouca integração, mas já atinge alguns dos requisitos legais, podendo dispensar o papel em alguns casos;
- c) Registro Médico Eletrônico (*Electronic medical record*): consiste em um modelo interdepartamental, reunindo os requisitos legais para confidencialidade, segurança e integridade dos dados;
- d) Registro Eletrônico do Paciente (*Electronic patient record*): sistemas neste nível interligam todas as informações do paciente, inclusive dados fora da instituição (interinstitucional). Para se chegar a este estágio, é necessário uma maneira de identificar o paciente de forma unívoca e nacional;
- e) Registro Eletrônico de Saúde (*Electronic health record*): neste último nível, além das características evolutivas dos anteriores, a responsabilidade de manter o prontuário é dividida entre profissionais de saúde e paciente.

O S-RES é um sistema pelo qual os registros eletrônicos de saúde são criados, manipulados, armazenados e recuperados. O S-RES inclui pessoas, dados, regras, procedimentos, dispositivos de armazenamento e processamento, meios de comunicação e manutenção.

A norma ISO/TR 20.514:2005 (*Health Informatics – Electronic Health Record – Definition, scope and context*), define Registro Eletrônico de Saúde – RES – como repositório de informação a respeito da saúde de indivíduos, numa forma processável eletronicamente. A norma ISO/TR 20.514:2005 define também Sistema de Registro Eletrônico de Saúde – S-RES – como sistema para registro, recuperação e manipulação das informações de um Registro Eletrônico de Saúde (ISO/TR, 2005 apud SBIS, CFM, 2008).

2.3 S-RES: VANTAGENS E DESVANTAGENS

Um prontuário em papel bem estruturado pode ser eficiente. Entretanto, atualmente, não é suficiente para atender todas as necessidades dos médicos e pode limitar a atuação do médico em situações de tomadas de decisões. Algumas desvantagens do prontuário em papel são (MARIN; MASSAD; AZEVEDO NETO, 2003):

- a) seu acesso é restrito à uma localidade e pode estar indisponível ou mesmo ser perdido;
- b) seu conteúdo é livre e algumas vezes ilegível, incompleto e com informações ambíguas;
- c) para estudos científicos, o conteúdo requer transcrição, o que predispõe ao erro e;

d) anotações manuscritas em papel não podem disparar lembretes e alertas automáticos aos profissionais.

Um S-RES tem diversas vantagens em relação ao Prontuário em Papel, tais como as listadas no Quadro 1 (SITTIG, 1999 apud MARIN; MASSAD; AZEVEDO NETO, 2003).

Vantagens do uso de um S-RES
<p>Acesso remoto e simultâneo - Vários profissionais podem acessar um mesmo prontuário simultaneamente e de forma remota. Com a possibilidade de comunicação segura via redes de computadores, os médicos podem acessar e editar, seguindo critérios de ética médica e legislação apropriada, os prontuários de seus pacientes a partir de qualquer ponto de acesso.</p>
<p>Legibilidade - Registros feitos à mão são notoriamente difíceis de ler. Os dados na tela ou mesmo impressos são mais acessíveis do que os manuscritos.</p>
<p>Segurança dos dados - Frequentemente, alguns usuários preocupam-se com a possível perda de dados devido ao mau funcionamento de sistemas. Entretanto, num sistema bem projetado, com esquemas seguros de backup e planos contra desastres, o S-RES é confiável no que diz respeito a dados e perdas de dados.</p>
<p>Confidencialidade dos dados do paciente - O acesso ao prontuário pode ser restrito e monitorado automaticamente, com cada usuário tendo um nível de acesso específico. Além disso, registros de auditoria podem ser utilizados para a detecção de acessos não-autorizados.</p>
<p>Flexibilidade do layout dos dados - Os usuários podem usufruir de formas diferentes para a visualização e entrada dos dados, vê-los em ordens distintas, tais como cronológica crescente ou decrescente, orientada ao problema ou orientado à fonte.</p>
<p>Integração com outros sistemas de informação - Os dados do paciente, em formato eletrônico, podem ser interligados para armazenar as informações localmente ou via rede, por exemplo, em outro computador.</p>
<p>Captura automática de dados - Dados fisiológicos podem ser capturados automaticamente de monitores nos leitos, analisadores de laboratório e equipamentos de imagens localizados em qualquer parte do hospital, evitando-se, por exemplo, erros de digitação.</p>
<p>Processamento contínuo dos dados - Os dados são estruturados e codificados de forma não-ambígua. Os softwares podem continuamente checar e filtrar os dados em busca de erros, sumarizar e interpretar, bem como emitir alertas e lembretes para os médicos.</p>
<p>Assistência à pesquisa - Os sistemas podem pesquisar em texto livre, palavras-chave, bem como sendo armazenados de forma estruturada, encontrar dados específicos para determinar se um item em particular foi registrado ou não; possibilitando pesquisas coletivas e facilitando levantamentos estatísticos.</p>
<p>Diversas modalidades de saídas de dados - Os dados podem ser apresentados para os usuários de diversas formas: voz, impresso, ou por e-mail, por exemplo. Além disso, comandos podem ser enviadas para equipamentos tais como bombas de infusão. Alarmes podem ser disparados. Também, imagens podem ser processadas, com apresentações em forma tridimensional, dentre outras aplicações.</p>
<p>Construção de diversos tipos de relatórios - Os dados podem ser impressos usando-se uma variedade de fontes, cores e tamanhos ajudando a chamar a atenção dos médicos para dados mais importantes. Além disso, imagens podem ser impressas em conjunto com os dados, criando-se figuras mais completas, que ilustram e ajudam a diagnosticar a doença do paciente.</p>
<p>Os dados estão sempre atualizados - Em um S-RES integrado, os dados podem estar disponíveis para consulta em todos os seus pontos de acesso.</p>

Quadro 1. Vantagens do uso de um S-RES.

Fonte: SITTIG, (1999) apud MARIN; MASSAD; AZEVEDO NETO; (2003).

O Prontuário Eletrônico também possui desvantagens, como as apresentadas por Barnett e McDonald (1990) no Quadro 2.

Desvantagens do uso de um S-RES
<p>Necessidade de grande investimento de hardware, software e treinamento: Deve-se ter uma quantia considerável de dinheiro para compra dos dispositivos móveis, treinamento dos médicos e enfermeiros no uso do prontuário e compra dos softwares específicos para cada dispositivo móvel;</p> <p>Resistência dos usuários quanto ao uso dos procedimentos informatizados: Os profissionais em saúde estão mais acostumados com o uso do papel e encontram certa dificuldade com a utilização de sistemas informatizados;</p> <p>Demora no desenvolvimento e na efetiva utilização, que são os resultados do investimento: O desenvolvimento de um Registro Eletrônico é complexo e custa muito tempo e dinheiro. Os resultados iniciais, em sua maioria, não são os esperados pelos médicos, assim tornando mais demorada sua utilização;</p> <p>Estar sujeito a falhas tanto de hardware como de software: sistema torna-se inoperante por minutos, horas ou até dias, que se traduz em informações indisponíveis;</p> <p>Dificuldade para completa coleta de dados: Os dispositivos móveis podem travar com uma coleta muito grande de dados, prejudicando no armazenamento dos dados dos pacientes.</p>

Quadro 2. Desvantagens do uso de um S-RES
 Fonte: BARNETT, G; MCDONALD, C. (1990).

Em 1969 foi proposto pelo Dr. Lawrence Weed uma abordagem denominada Registro Médico Orientado ao Problema (*Problem-Oriented Medical Record - POMR*), no qual a organização interna do prontuário é indexada por cada problema médico do paciente. Ou seja, anotações, terapêuticas, diagnósticos, dentre outros são relacionados a um problema específico de saúde. Assim, constitui-se uma lista de problemas hierarquicamente distribuídos na forma de uma árvore, na qual cada problema é um ramo (WEED, 1968). Neste modelo, a estrutura do S-RES pode ser dividida em informações do paciente e lista de problemas, que constitui uma tabela de conteúdos do registro.

Cada registro (evolução, resultado de exame, entre outros) é indexado por problema, de acordo com a estrutura SOAP - *Subjective Objective Assessment Plan* (VAN GINNEKEN; MOORMAN, 1997): *Subjective* - sintomas do paciente, ou seja, informações associadas às queixas do paciente; *Objective* - referente aos sinais observados pelo médico, ou

seja, informações que o médico recolhe, tais como sintomas, exames, dentre outras; *Assessment* - resultados de exames e conclusões, tal como um diagnóstico, com base em informações prévias e exames e; *Plan* - conduta, como por exemplo, um tratamento, prescrição de medicamentos, cirurgia ou medidas à longo prazo.

Este modelo de atendimento utiliza a informação e a integração como elementos essenciais de organização. Neste aspecto, surge uma estrutura computacional chamada de Sistema de Registro Eletrônico de Saúde (S-RES), apresentando uma proposta para unir os diferentes tipos de dados produzidos em vários formatos, feitos por diferentes profissionais da equipe de saúde de épocas diferentes, em distintos locais (MARIN; MASSAD; AZEVEDO NETO, 2003).

Na Figura 1 é apresentado o modelo de Registro Médico Orientado ao Problema (RMOP).

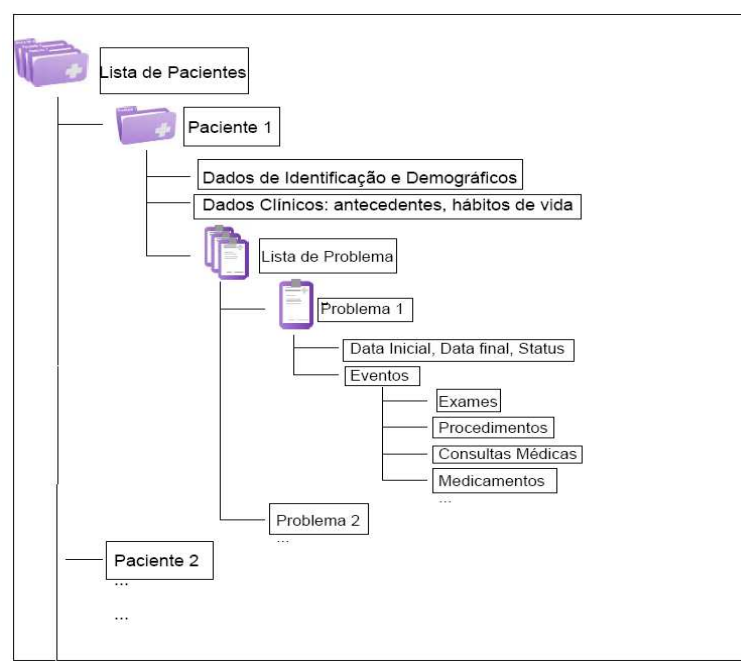


Figura 1: Modelo RMOP.

Fonte: COSTA, C. (2001) apud GOULARTE, F. (2006).

Com o crescimento do número de S-RES que programaram esse modelo, começaram a aparecer algumas limitações, tais como: subproblemas que não são

contemplados pela estrutura clássica do modelo e anotações de consultas médicas (COSTA, 2001 apud GOULARTE, 2006).

Outro problema que afeta esse modelo é que a entrada de dados pelos usuários é lenta, devida à sua estrutura em que prevalece a identificação de dados já inseridos e tratamentos realizados previamente. Por esse motivo, novos modelos têm surgido para contornar esses problemas. Alguns modelos conduzem a subdivisões diferentes do SOAP, tais como a O-HEAP - Orientação, História, Exame, Avaliação e Planejamento (MEYERS; MILLER; NAEYMI-RAD, 1998 apud COSTA, 2001). Outras subdividem a lista de problemas e contornam deficiências referentes à interseção de eventos entre problemas (PKC Corporation, 1998 apud COSTA, 2001). Entretanto, todas essas modificações mantêm a base da estrutura do RMOP.

2.5 PADRÕES DE REGISTRO E TRANSMISSÃO DE DADOS EM SAÚDE

Com inúmeras entidades criando e usando diversas formas de S-RES, que tipicamente requerem comunicação de dados entre sistemas e equipamentos médicos, é fundamental a adoção de padrões para garantir interoperabilidade entre sistemas.

Atualmente, há vários padrões em uso. Há tipicamente padrões para vocabulário, conteúdo, comunicação, imagens, objetos, dentre outros. De acordo com as características, os padrões para a área da saúde podem ser classificados da seguinte forma (BLAIR, 2001 apud COSTA, 2001):

- a) Identificação - para pacientes (*Social Security Number* - nos Estados Unidos, Cartão Nacional de Saúde – No Brasil), médicos (Número no Conselho Regional de Medicina), entre outros;

- b) Comunicação - padrão para mensagens entre sistemas (HL7, XML, entre outros);
- c) Conteúdo e Estrutura - padronização para registro clínico (DATASUS, ABRANGE);
- d) Representação dos dados clínicos (CID, SNOMED, LOINC, entre outros.);
- e) Confidencialidade, Segurança e Autenticação;
- f) Indicadores de qualidade, conjunto de dados e diretrizes.

Segundo Costa (2001) os padrões são necessários também para viabilizar o uso de sistemas de apoio à decisão e sistemas de alerta, bem como são indispensáveis para a interoperabilidade entre os sistemas.

A adoção de padrões no desenvolvimento e uso de um S-RES viabilizarão integrações com sistemas distintos. Padrões e normas são referenciais que devem ser seguidos para a correta organização e acesso às informações em sistemas de saúde (MARIN; MASSAD; AZEVEDO NETO, 2003).

A elaboração de padrões é tarefa de organizações espalhadas em todo o mundo, tais como a *International Standards Organization (ISO)*, o *American National Standards Institute – Healthcare Informatics Standards Board (ANSI-HISB)* e o *Comité Européen Normalization (CEN)*. Este último contém um grupo de trabalho, na União Européia, denominado CEN/TC251 (*CEN Technical Committee 251*) trabalhando em padronização no campo de Tecnologia de Informação e Comunicação em Saúde (*Health Information and Communications Technology - ICT*). No Brasil, a Associação Brasileira de Normas Técnicas (ABNT) é o órgão responsável pela normatização em diversas áreas. Como parte de estudo desse projeto, apresentamos alguns dos padrões de comunicação de informações que, atualmente, estão em uso no desenvolvimento de prontuários.

2.5.1 Padrão *OpenEHR*

A Fundação *OpenEHR* é uma organização sem fins lucrativos cujo objetivo é transformar em realidade registros eletrônicos interoperáveis, permitindo assim que as informações estejam disponíveis para a área da saúde da melhor forma possível (FRANÇA; NARDON; NAVES, 2008).

O padrão trata de um sofisticado modelo orientado a objetos que incorpora tipagem forte com tipos de dados muito robustos para representar a informação em saúde, como duração, período e datas parciais, medidas e dados qualitativos. O referencial teórico do modelo *OpenEHR* está baseado no prontuário orientado a problemas, no modelo hipotético-dedutivo do diagnóstico médico e nas teorias cognitivas do raciocínio médico (NEIRA et al, 2008).

O modelo *OpenEHR* está baseado numa ontologia de conceitos representados por arquétipos organizados nas seguintes categorias (NEIRA et al, 2008):

- a) *Compositions* de arquétipos temáticos - documentos clínicos como: sumário de alta, avaliação pré-anestésica, e cuidados de enfermagem, entre outros;
- b) *Sections* - Arquétipos organizacionais utilizados para definir a navegação no RES: história, exame físico, evoluções;
- c) *Observations* – registro de dados mensuráveis ou observados: pressão arterial, sintomas, peso;
- d) *Evaluations* – registro de avaliações clínicas: avaliação de risco anestésico, efeito adverso;
- e) *Instructions* – registro do início de processo de “*workflow*”: prescrição médica, solicitação de exames;

f) *Actions* – registro da atividade clínica: administração de medicamentos ou realização de procedimentos. *Actions* complementam as *instructions* e registram seus estados, tais como: “*completed*” or “*cancelled*”.

O padrão desenvolvido pela fundação é centrado num conceito de arquétipos. Arquétipos são descritos como um modelo formal e reutilizável de um conceito do domínio. Assim, se um conceito é representado por um arquétipo, este conceito pode ser reutilizado nos vários cenários em que se aplica. Os arquétipos são definidos em uma linguagem computável chamada de ADL (*Archetype Definition Language*) (FRANÇA; NARDON; NAVES, 2008).

Diferentes arquétipos podem ser agrupados em um *template*. Um *template* geralmente é usado para definir uma ficha clínica, ou um formulário que será preenchido. Nele é possível determinar os arquétipos que serão utilizados, as ocorrências que devem existir de cada arquétipo, elementos que, dentro de um arquétipo, serão utilizados, entre outras definições (FRANÇA; NARDON; NAVES, 2008).

Quando as informações descritas pelos *templates* e arquétipos são instanciadas, ou seja, quando os dados clínicos reais são informados, através do preenchimento de um formulário, por exemplo, dá-se origem a um documento chamado de *composition*. São elas que efetivamente contém os dados clínicos que formam as diversas fichas clínicas que compõem o prontuário do paciente (FRANÇA; NARDON; NAVES, 2008).

O padrão que define como uma *composition* deve ser representada é baseado em XML. Portanto, é possível exportar informações para o formato de *composition*, mesmo que a aplicação original que coletou as informações não tenha sido baseada em arquétipos e *templates* (FRANÇA; NARDON; NAVES, 2008).

O *openEHR* também apresenta algumas deficiências, tais como (LOPES; PETRY; WANGENHEIM, 2006):

- a) resistência dos patrocinadores para incorporar componentes *openEHR*, devido principalmente ao custo e risco envolvidos;
- b) mostra ser difícil de implementar, por isso não são encontradas implementações em média e alta escala;
- c) ainda é considerado instável, resultante de atividades de pesquisa e desenvolvimento, por isso, sujeito a significantes mudanças;
- d) o suporte é prestado por poucos indivíduos - chave;
- e) as especificações do *openEHR* não são aceitas como um SDO.

2.5.2 Padrão TISS

Esse padrão foi criado pela Agência Nacional de Saúde Suplementar (ANS), visando à padronização da troca de informações entre operadoras de plano privados de saúde e os prestadores de serviços de saúde. Para sua criação, foram estudados padrões internacionais de trocas de informações, as quais se destacam o HL7 v3, *OpenEHR* e ISO-TC215 (BRASIL, 2007).

De acordo com Brasil (2007), a padronização e a troca eletrônica de informações em saúde suplementar trazem inúmeros benefícios, entre os quais:

- a) aprimoram a comunicação entre os atores do setor;
- b) reduzem o uso de papel, agilizando o acesso do beneficiário aos serviços de saúde;
- c) facilitam a obtenção de informações para estudos epidemiológicos e definição de políticas em saúde;
- d) favorecem a realização de análise de custos e benefícios de investimentos na área de saúde;

- e) reduzem custos administrativos;
- f) melhoram a qualidade da assistência à saúde;
- g) possibilitam comparações e análises de desempenho institucional implicando a otimização de recursos e aumento da qualidade de gestão.

O padrão TISS usa a linguagem de marcação *XML / Schema* e normas de segurança vigentes editadas pelo Conselho Federal de Medicina.

Nesse esquema, a troca de informações está definida como mensagens TISS. Ela é composta de cabeçalho, corpo de mensagem e epílogo. O cabeçalho descreve a origem e o destino da mensagem. O corpo contém as transações específicas do padrão TISS e o epílogo contém o fechamento da mensagem com o cálculo do *hash*.

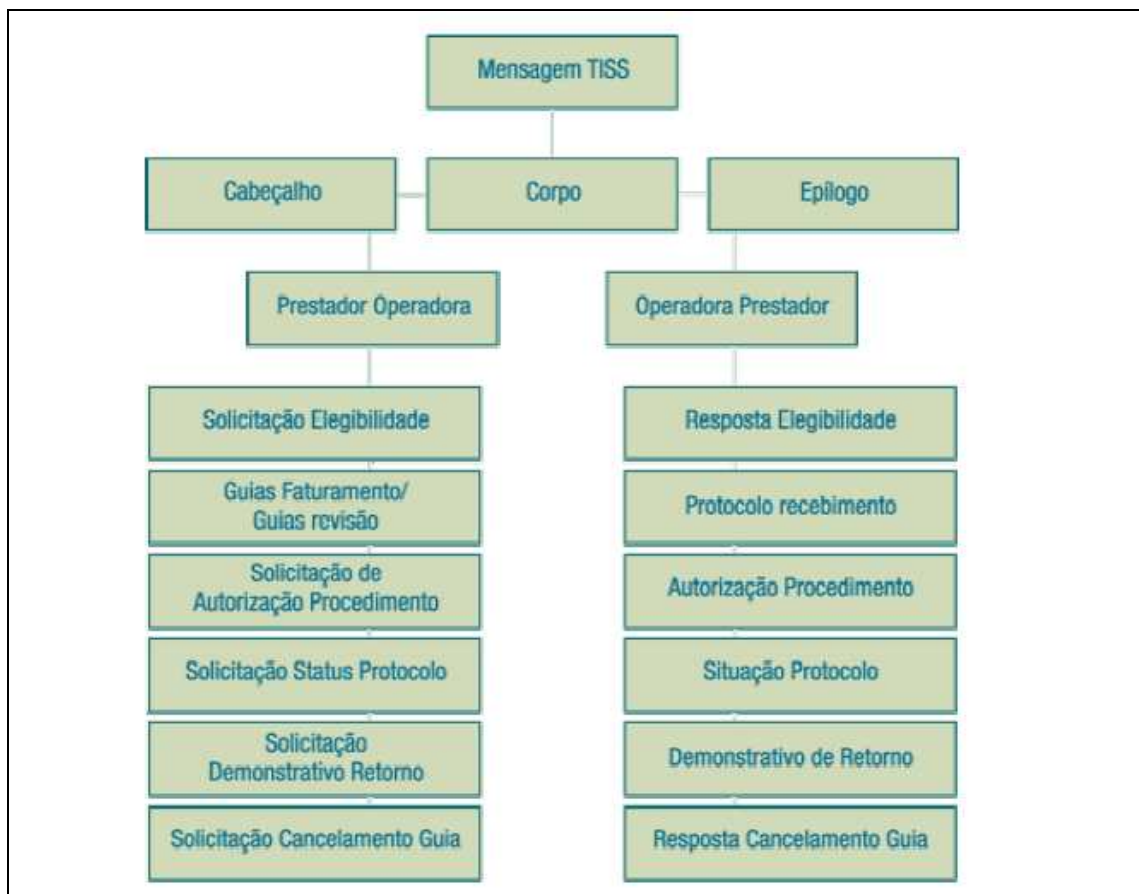


Figura 2: Demonstração de envio de mensagem utilizando o padrão TISS.
Fonte: BRASIL (2007).

2.5.3 Padrão HL7 (*Health Level Seven*)

No modelo ISO (*International Standards Organization*) as funções de comunicação são divididas em 7 camadas. O termo L7 refere-se ao nível 7 do modelo ISO. O padrão HL7 é focado no nível da aplicação.

O HL7, controlado pelo ANSI-HISB, é um padrão internacional para troca, gerência e integração de informações eletrônicas em sistemas de atenção a saúde, em nível de aplicação disponibilizando aspectos referentes à composição dos dados utilizados em transferências, bem como as maneiras de efetuar tais trocas e o reconhecimento de erros comuns entre as aplicações, para se utilizado dentro de e entre instituições de saúde visando aumentar a eficiência e a eficácia nas demandas de atenção à saúde para benefício da população. (MARIN; MASSAD; AZEVEDO NETO, 2003). O HL7 é baseado em objetos com foco na interoperabilidade entre sistemas que enviam, recebem e acessam dados de pacientes.

2.5.4 Classificação Internacional de Doenças (CID)

A Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde - CID (*International Statistical Classification of Diseases and Related Health Problems - ICD*) é uma catalogação de diversas doenças, sinais, sintomas, aspectos anormais, queixas, dentre outros, que é representada por meio de códigos únicos, visando padronização das informações em prontuários de paciente. A CID vem sendo modificada ao longo do tempo. Em um S-RES específico para uma UTI, esta classificação deve ser adotada na realização dos diagnósticos e nas causas de óbito (CBCD, 2008).

2.5.5 XML para Comunicação

O padrão XML (*eXtensible Markup Language*) possibilita a troca de documentos eletrônicos na *Web* entre tecnologias distintas. Foi elaborado pelo *World Wide Web Consortium* (W3C), que é a entidade que coordena iniciativas na *Web* (W3C, 2008). O XML é um padrão com um conjunto de regras e convenções para definir formatos de textos estruturados e identificar os dados que compõe documentos estruturados e dados na *Web*.

Assim, qualquer tipo de documento poderá ser codificado e “assinado” pelo autor XML, disponibilizando *tags* específicas que identificam que estes foram assinados. A utilização de XML assinado é de suma importância na troca de informações eletrônicas, pois possibilita identificar a origem das informações que são recebidas, bem como o autor destas.

2.6 A ÉTICA NA UTILIZAÇÃO DO S-RES E ASPECTOS LEGAIS

Toda e qualquer informação médica requer a mais absoluta segurança. Dados pessoais devem ser salvos em bancos de dados com a máxima segurança. O profissional de saúde, bem como os profissionais responsáveis pela segurança e manuseio das informações sobre os pacientes, são regidos pelos ditames da lei e podem ser processados, por utilização ou difusão indevida das informações, segundo o Código Civil, Código Penal e Código do Processo Penal. No caso de médicos, ou outros profissionais, com nível superior, podem sofrer as sanções determinadas pelos Códigos de Ética das Respectives Profissões (MARIN; MASSAD; AZEVEDO NETO, 2003).

Todas as profissões da área da saúde são regulamentadas por leis, normas e códigos que consagram os princípios éticos definidos primária e primeiramente para a medicina e

estabelecem limites à atuação de seus integrantes (MARIN; MASSAD; AZEVEDO NETO, 2003).

Em relação às questões éticas os regulamentos existentes, são suficientes para preservar o segredo profissional, devendo garantir-se, sempre e permanentemente os cuidados devidos à monitorização e fiscalização do cumprimento das regras.

Em referência ao S-RES, há normas presentes nos códigos de ética profissionais, que delimitam obrigações e deveres como também sanções para o uso inadequado ou indevido aos profissionais envolvidos diretamente com o S-RES.

Os dados e informações que paciente fornecem durante a consulta, por meio de exames e procedimentos médicos são de propriedade do paciente² (COSTA, 2001).

Os legisladores brasileiros estão procurando produzir leis que sejam coerentes e adequadas às necessidades em termo de acesso aos dados a também à privacidade dos mesmos. Por último, é importante salientar que a proteção aos dados é um direito humano fundamental o qual está relacionado com o direito à privacidade (D'ORNELLAS; ROCHA, 2002).

² Anteriormente, acreditava-se que estas informações eram do médico ou da instituição, por isso o termo “prontuário médico”. Porém, tanto instituições de saúde como seus profissionais somente tem direito de usar as informações contidas no prontuário para fins de assistência ao paciente. Além disso, devem respeitar o sigilo destas, considerando-se que a instituição e os profissionais da saúde são fiéis depositários do prontuário (COSTA, 2001).

3 COMPUTAÇÃO MÓVEL

A Computação Móvel envolve o uso de DM, provendo mobilidade no acesso aos serviços de rede. Os DM ampliam o domínio de aplicações por meio das tecnologias de redes sem fio e dos sistemas distribuídos, garantindo mobilidade em casos em que a conexão/comunicação é requerida (CRISPIM. JUNIOR; FERNANDES, 2006).

3.1 DISPOSITIVOS MÓVEIS

Atualmente, os dispositivos de computação móvel, chamados de Dispositivos Móveis - DM, possuem dimensões e, capacidade de armazenamento e processamento reduzidos – quando comparados aos computadores tipo *desktop* e *notebooks*. Os DM podem ser levados para qualquer lugar provendo mobilidade, comunicação, acesso a informações, transferências de dados, dentre outras facilidades aos usuários.

O uso dos DM vem crescendo graças a junção de tecnologias de comunicação, tais como acesso a Internet sem fio, telefonia móvel celular e disponibilização de aplicações que incluem agendas eletrônicas, editores de texto, planilhas eletrônicas, navegadores, bancos de dados, dentre outras (PRÁ JUNIOR, 2006).

Para profissionais que requerem mobilidade no seu cotidiano e necessitam de informações atualizadas bem como a edição destas, independente da sua localização, os DM vêm se tornando ferramentas cada vez mais frequentes. É o caso de profissionais da área médica. Há profissionais que realizam seus atendimentos em postos fixos que possibilitam o uso de computadores *desktop* para acesso aos S-RES. Mas há também os profissionais de saúde que estão em constante movimento em seus ambientes e necessitam cada vez mais de ferramentas que acompanhem essa rotina (CRISPIM. JUNIOR; FERNANDES, 2006).

Vários aspectos devem ser considerados na utilização de DM, bem como no desenvolvimento de soluções para estes. A seguir serão apresentados alguns dos aspectos relevantes (KOTZ; CHEN, 2006 apud CRISPIM JUNIOR; FERNANDES, 2006; PEKUS, 2002):

- a) Dimensões - além de leves e de simples manuseio, podem ser facilmente transportados;
- b) Interface - *display* e teclas de tamanho reduzido;
- c) Características do ambiente – ocorrem maior números de desconexões e interferências devido à mobilidade;
- d) Consumo de energia - mobilidade dos DM implica necessidade de fontes de energia próprias, tipicamente baterias, cujo consumo e tempo de recarga são fatores consideráveis;
- e) Capacidade de Processamento - esses dispositivos possuem reduzida capacidade de processamento e memória que não devem ser desprezadas ao se projetar à aplicação;
- f) Segurança - redes sem fio são sujeitas ataques, sendo mais suscetíveis à interceptação de dados, já que seu meio de propagação é o ar.

Conforme Maluta (2001) um DM deve ser leve, de manuseio simplificado, pequeno, e portátil. Como exemplos de dispositivos móveis tem-se telefones celulares, PDA, *smartphones* e *Tablet PCs*.

3.1.1 TELEFONES MÓVEIS CELULARES

Os telefones móveis celulares são vastamente usados atualmente. Os sistemas de telefonia móvel celular possibilitam que os usuários possam utilizar os serviços de

comunicação nas áreas de cobertura das operadoras (MOURÃO, 2004 apud PRÁ JUNIOR, 2006).

Recentemente a infra-estrutura da telefonia celular, bem como as tecnologias incorporadas aos aparelhos telefônicos, veio se aprimorando. Atualmente há numerosos aparelhos celulares. Alguns deles trazem as funções básicas, tais como realizar chamadas telefônicas e mandar mensagens. Outros trazem, adicionalmente, funções avançadas, tais como fotografar, rodar arquivos mp3 e acessar a Internet (MOURÃO, 2004 apud PRÁ JUNIOR, 2006).

A maioria dos usuários dos telefones móveis celulares o utiliza para sua função básica. Um fator que dificulta o uso de outras funções nestes aparelhos, quando disponível, é sua interface reduzida e compacta. Por exemplo, possui poucas teclas para representar extensos alfabetos de letras, números e símbolos. Outro fator que prejudica o uso de outras funções é a incompatibilidade de padrões de telas e menus entre dispositivos de distintos fabricantes (PRÁ JUNIOR, 2006).

Segundo Murakami (2003) os telefones móveis celulares são divididos cronologicamente segundo os protocolos de comunicação utilizados. A primeira geração de telefones móveis celulares (1G) possuía comunicação analógica e não possibilitava a transmissão de dados digitais. É uma tecnologia em desuso. A segunda geração (2G) usa comunicação digital e é atualmente bastante difundida no país. Celulares 2G possibilitam, além da comunicação por voz, transmissão de dados digitais a baixa velocidade (14,4 Kbps), acesso a Internet (WAP) e envio de mensagens SMS. Os protocolos mais utilizados são o *Time Division Multiple Access* (TDMA), *Code Division Multiple Access* (CDMA) e *Global System for Mobile communications* (GSM). Esses protocolos possibilitam uma taxa de transmissão de cerca de 10 Kbps (MURAKAMI, 2003). A geração 2,5G utiliza tecnologia de transmissão por pacotes, como *General Packet Radio Service* (GPRS) e CDMA 1x. Alguns

serviços foram incorporados, tais como acesso permanente à *web*, possíveis pelo aumento das velocidades, e tarifação por volume de dados. A cada canal de voz, há um canal de dados associado. Porém, não é possível acessar a *web* e receber ligações simultaneamente (PALUDO, 2001). A recente terceira geração de telefonia móvel celular (3G) no país utiliza tecnologia *Wideband Code Division Multiple Access* (W-CDMA³). Esse serviço oferece além de boa qualidade na transmissão de voz, suporte a diversos serviços multimídia, tais como transmissão de voz e vídeo, imagens, músicas, possibilidade de compartilhamento de aplicações tais como jogos, além de comunicação de dados em banda larga (MURAKAMI, 2003).

3.1.2 PDA

O PDA - *Personal Digital Assistant* - é um DM de dimensões reduzidas que incorpora funções típicas dos computadores tais como armazenamento, processamento e recuperação de informações e conectividade a redes de computadores.

Atualmente, os PDA já substituem *notebooks* em algumas aplicações que requerem mobilidade e agilidade, que são alcançadas pela autonomia de uso (bateria), facilidade de deslocamento e peso reduzidos (BERNAL, 2002 apud PRÁ JUNIOR, 2006).

Nos PDAs podem ser utilizados os mais variados sistemas operacionais, tais como o *Linux*, *Palm OS* e *Windows CE*.

O *Linux* é um sistema operacional de código aberto, geralmente utilizado em servidores de aplicações e *desktops*. Este sistema operacional é uma alternativa aos sistemas operacionais proprietários que vêm instalados nos PDAs (JANSEN; AYERS, 2004 apud PRÁ JUNIOR, 2006). Atualmente, existem poucos PDAs disponíveis com o *Linux* instalado como

³ É um padrão de interface rádio, entre o terminal celular e a Estação Rádio Base, com velocidade de transmissão de dados a mais de 2 Mbps.

sistema operacional, mas com o crescimento das distribuições e também o empenho dos usuários da comunidade *Linux* em pouco tempo esta situação será revertida (MOURÃO, 2004 apud PRÁ JUNIOR, 2006).

O *PalmOS* é um sistema operacional multitarefa. Ele fica armazenado na *Read Only Memory* (ROM), enquanto os aplicativos e arquivos do usuário ficam na *Random Access Memory* (RAM). Ele organiza as memórias para que fique transparente o funcionamento do PDA quando o usuário for inserir um cartão de memória adicional no dispositivo (JANSEN; AYERS, 2004 apud PRÁ JUNIOR, 2006).

O *Windows CE* é o sistema operacional da Microsoft. Ele é desenvolvido tipicamente para dispositivos móveis. A implementação das suas aplicações utiliza linguagem C++ e as APIs do sistema operacional (FIGUEIREDO, 2005 apud PRÁ JUNIOR, 2006).



Figura 3: Palm m130
Fonte: MURAKAMI (2003).

3.1.3 SMARTPHONES

Smartphone é um telefone celular com as funções típicas de PDA incorporadas tais como *e-mail* e aplicativos relacionados ao sistema operacional. Isso possibilita o uso das funcionalidades de um PDA na infra-estrutura de comunicação da rede de telefonia celular (FIGUEIREDO; NAKAMURA, 2003). Em relação à comunicação de dados, os *smartphones*

suportam múltiplas formas de conexão (CDMA, *Wi-Fi*, *Bluetooth*, GPRS, entre outras).

A Figura 4 ilustra dois modelos de *smartphones*, um deles usado neste trabalho.



Figura 4: (a) Nokia E61; (b) Nokia E71.
Fonte: NOKIA (2009).

No caso do Modelo E61, da Nokia, que foi adotado neste trabalho, seu Sistema Operacional é o Symbian 9.1. Possui suporte às tecnologias/protocolos de comunicação *WAP*, *GPRS/EDGE*, *IrDA*, *Bluetooth*, *IEEE 802.11*, entre outras.

3.1.4 *TABLET PC*

Os *Tablet PC* são computadores com tela sensível ao toque, equivalentes a *laptops* ou *notebooks*. Há tipicamente dois tipos de *Tablet PC*. O primeiro, chamado de conversível, possui formato semelhante ao de um *notebook*. Entretanto, quando aberto, possibilita que a tela seja rodada em 180° e fechada, tomando a forma de uma prancheta. O segundo tipo, menos comum, possui um teclado destacável (MURAKAMI, 2003). A Figura 6 apresenta um exemplo de *Tablet PC*.



Figura 5. HP Compaq Tablet PC tc1100
Fonte: MICROSOFT, 2009.

4 TECNOLOGIAS ASSOCIADAS AO DESENVOLVIMENTO DE APLICAÇÕES E USO DE DISPOSITIVOS MÓVEIS

Nas últimas décadas os DM têm passado por mudanças reduzindo dimensões, modificando formas de entradas de dados, incluindo telas sensíveis ao toque. Atualmente tem-se pequenos dispositivos pessoais tais como *Smartphone* capazes de transmitir e receber voz e dados. Isso abre oportunidades para operadoras, fabricantes e desenvolvedores, abrangendo setores tais como entretenimento, por meio de jogos, músicas e “*ringtones*”, serviços de segurança, com transmissão de imagens e localização, comércio, usando comunicação entre máquinas, propaganda com o SMS, “*Short Message Services*”, e incrementada com o surgimento do MMS, “*Multimedia Message Services*”, dentre outros (ARAUJO, 2006).

Nessa linha de desenvolvimento de aplicações para DM, têm sido usadas diversas tecnologias, dentre as quais se destacam *Python*, *Perl*, *C++*, *Java*, entre outras.

4.1 TECNOLOGIA JAVA

A tecnologia Java define uma linguagem de programação e uma plataforma de software, cujo objetivo é a portabilidade do código em que o mesmo software possa ser executado em distintos sistemas operacionais (SUN, 2007). A linguagem de programação Java possibilita que os programas sejam executados em computadores *desktop*, servidores e diversos outros dispositivos, dentre eles DM tais como PDA e celulares.

Os programas Java geram códigos compilados, denominados *bytecodes*, que são interpretados através de uma *Java Virtual Machine* (JVM). A JVM interpreta os *bytecodes* e gera o código binário para o sistema operacional sobre o que o programa será executado.

Conforme Kuhnen (2003) apud Schaefer (2004), Java apresenta vantagens quanto ao desenvolvimento de aplicações para dispositivos móveis, dentre as quais destacam-se:

- a) Orientação a objetos;
- b) Portabilidade e independência de plataforma - possibilitam sua utilização em qualquer sistema operacional que disponha de Máquina Virtual Java;
- c) *Multithreading* - possibilita desenvolver aplicativos que executam múltiplas tarefas em paralelo;
- d) Robustez e Segurança - Java tem capacidade interna de impedir a corrupção de memória, administra os processos de alocação de memória e a checagem de limite de *array*.

A plataforma Java é dividida em três especificações (SUN, 2007):

- a) *Java 2 Platform, Standard Edition* - apresenta recursos para desenvolvimento de aplicações *desktop*, como, por exemplo, *API* para desenvolvimento de interfaces gráficas;
- b) *Java 2 Platform, Enterprise Edition* - possibilita a construção de aplicações para servidores. É voltada para aplicações corporativas, disponibilizando, dentre outras, ferramentas para construção de aplicações *Web* e;
- c) *Java 2 Platform, Micro Edition* - tem recursos limitados; possibilita a construção de aplicações voltadas para DM, os quais, de modo geral, possuem tela e usabilidade reduzida e baixa capacidade de processamento.

A Figura 6 ilustra a plataforma Java em suas três especificações.

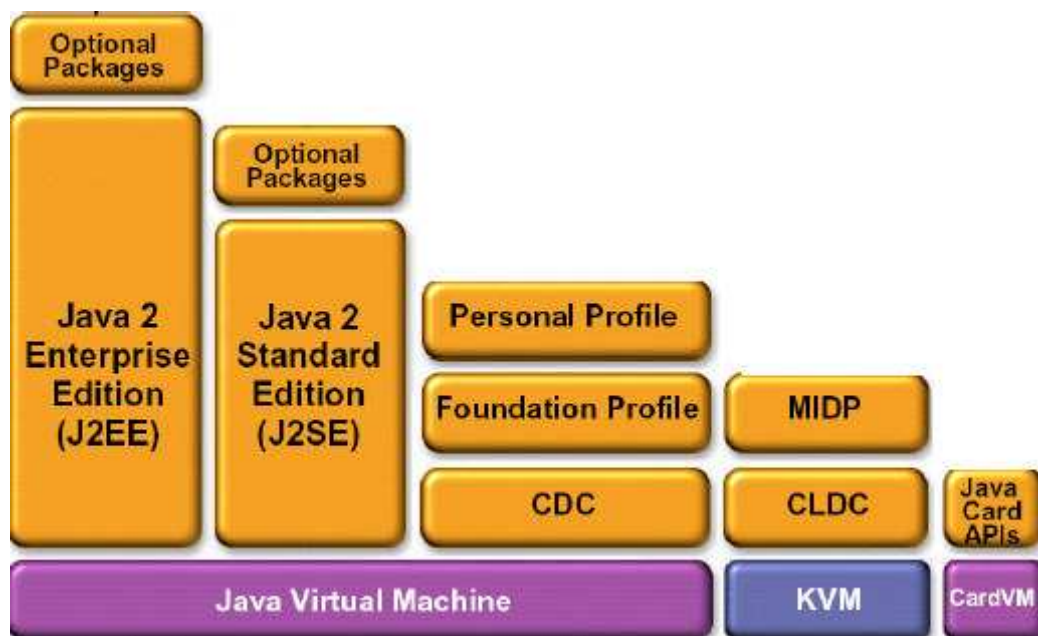


Figura 6: Plataformas Java.
Fonte: PALUDO, 2003

4.1.1 *Java Enterprise Edition*

O *Java Enterprise Edition* (J2EE) é a plataforma Java destinada ao desenvolvimento de aplicações corporativas, para as quais fornece um conjunto de serviços para componentes modulares e para o tratamento automático de detalhes, facilitando a programação. Dispõe de um conjunto de práticas e especificações, interfaces de programação de aplicação (API) e protocolos, destinadas a aplicações em camadas (OLIVEIRA, 2004). A divisão em camadas possibilita a organização dos componentes de acordo com sua funcionalidade. Componentes que interagem para objetivos afins podem ser agrupados em camadas, facilitando manutenção e expansibilidade dos sistemas (EVANS, 2004). Estas aplicações são executadas tipicamente em servidores.

A plataforma J2EE é orientada a objetos, fator importante para a eficiência de programação, promovendo a reutilização de código. Também é portátil, desde que o sistema operacional tenha uma máquina virtual Java em execução (OLIVEIRA, 2004). Possui vasta documentação, o que facilita a programação. A plataforma J2EE contém uma série de API,

cada uma com funcionalidades distintas. Dentre elas destacam-se: *Java Database Connectivity* (JDBC), *Enterprise JavaBeans* (EJB), *Java Server Pages* (JSP) e *Servlets*.

O Servidor de Banco de Dados do S-RES está implementado utilizando a tecnologia J2EE. Em uma requisição de informações, o sistema de S-RES, com auxílio do *framework SpringMVC*, recebe a requisição e, se necessário, realiza consultas ao banco de dados por meio do *framework Hibernate*. Após o processamento das informações obtidas, uma resposta é enviada à aplicação cliente, também com auxílio do *framework SpringMVC*. Na aplicação proposta nesse trabalho, são utilizados os *frameworks Hibernate* e *Spring*, cujas descrições são apresentadas na seção 4.4, bem como suas utilizações no S-RES, são apresentadas no Capítulo 6. A seguir, serão discutidas as API da tecnologia J2EE utilizadas no S-RES.

4.1.1.1 JDBC

O JDBC é uma API que possibilita conexão, consulta ou atualização com um banco de dados, usando SQL (*Structured Query Language*), executando e processando os resultados de consultas SQL ao banco de dados. Disponível no pacote `java.sql`, trata-se de um conjunto de interfaces e classes para o desenvolvimento de aplicações Java para executar ações em qualquer banco de dados em uma arquitetura cliente/servidor. Um sistema de banco de dados é acessado por meio de um *driver* JDBC que implementa a interface `java.sql.Driver`. Há *drivers* para praticamente todos os bancos de dados relacionais, sendo que o JDBC acessa esse gerenciador de *driver* para poder acessar os registros do banco de dados (CADENHEAD, 2003).

4.1.1.2 *Servlets*

Servlets são classes da linguagem de programação Java usadas para expandir as capacidades dos servidores que hospedam aplicações de acesso por meio de uma solicitação de resposta. As *servlets* são executadas em uma (JVM) em um servidor *Web* denominado *servlet container*.

Embora *servlets* possam responder a qualquer tipo de solicitação, são comumente usados para expandir aplicações hospedadas em servidores *web*. Para essas aplicações, tecnologia Java *Servlet* define classes de *servlet* HTTP-específicas (ARMSTRONG, 2005).

Tipicamente, as *servlets* são um substituto para scripts *Common Gateway Interface* (CGI). A *web* inicialmente foi utilizada para disponibilizar conteúdo estático. Da necessidade de gerar conteúdo dinâmico surgiram os programas CGI. A primeira tecnologia java que tinha a finalidade de prover conteúdo dinâmico foi a *Servlet*, que são programas java, às quais executam em associação com servidores *web*, por meio do modelo *request / response*, lidando com características típicas do protocolo HTTP, tais como os métodos GET e POST, *cookies* dentre outros. As *Servlets* não possuem interface gráfica e operam não apenas com servidores *web*, mas com vários tipos de servidores, uma vez que sua API não contém explicitamente especificações ao ambiente do servidor. Além disso, são carregadas apenas uma vez e, por serem *multi-thread*, podem atender a mais de uma solicitação de uma só vez. Esta peculiaridade que a torna mais rápida que um programa CGI comum. Scripts CGI tradicionais escritos em *Perl*, *C*, entre outros, têm a desvantagem em que um novo processo deve ser criado para cada chamada do script. A sobrecarga dos processos de criação e a gerência dos processos podem sobrecarregar o servidor.

As *Servlets* contornam este problema por meio da criação de uma linha para cada pedido, em vez de um processo inteiro. Um único processo para cada *servlet* é criado e, em

seguida, um pedido para o *servlet* provoca um tópico a ser criado. A Figura 7 ilustra um exemplo de arquitetura *servlet*.

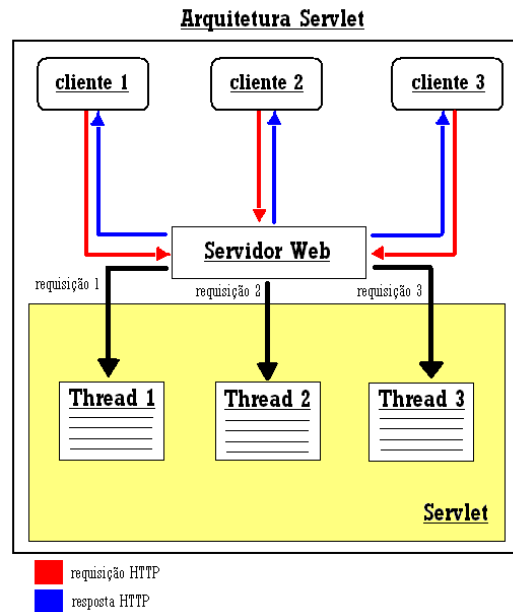


Figura 7: Arquitetura Servlet

Fonte: CDTC – Centro de Difusão de Tecnologia e Conhecimento, 2008.

As *servlets* têm acesso a todos os recursos da plataforma Java, por ser uma tecnologia Java. Desta forma as *servlets* têm acesso a todas as bibliotecas Java, o que ajuda a acelerar desenvolvimento de aplicações.

4.1.1.3 JSP

Java Server Pages (JSP) é uma tecnologia utilizada na criação de aplicações para a *web* com programação em Java. É uma linguagem que surgiu por volta do ano 2000, como uma alternativa da *Sun Microsystems* para competir com linguagens emergentes como *Active Server Pages* (ASP) – linguagem de autoria da *Microsoft* – e *Hypertext PreProcessor* (PHP). Por ser baseada na linguagem Java, tem a vantagem da portabilidade de plataforma, que possibilita a execução em vários sistemas operacionais. Um *JSP* faz as funções de um script em uma página *web* e é compilado como uma *servlet* em seu container (servidor). Uma página JSP contém dois tipos de texto: estático, representado por marcações HTML, WML, ou XML,

e elementos dinâmicos, que são as partes de uma página geradas por um *servlet*. Um código JSP pode ser escrito com um editor de texto HTML/XML comum, já que as páginas JSP estão compostas de código HTML/XML mesclado com etiquetas para programar scripts de servidor em sintaxe Java.

Os arquivos JSP têm extensão '.jsp', os quais, dentro das *tags* HTML, contém o código em java a executar no servidor. Antes que os arquivos sejam funcionais, o motor JSP realiza uma fase de tradução dessa página em uma *servlet*, implementado em um arquivo *class* (*Bytecodes* de Java). O motor das páginas JSP está baseado nos *servlets* de Java.

O JSP facilita na modificação de pequenos erros de digitação na apresentação da página, usando um *servlet* de dentro do conteúdo estático, como se fosse um conteúdo dinâmico. Assim, somente é recompilado um *servlet* e não o documento HTML inteiro. A Figura 8 ilustra a arquitetura JSP.

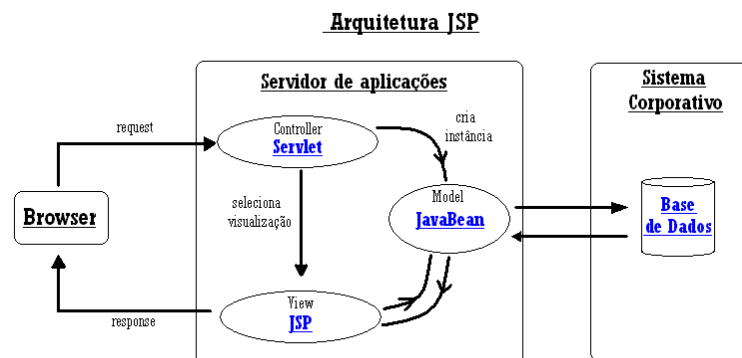


Figura 8. Arquitetura JSP.

Fonte: CDTC – Centro de Difusão de Tecnologia e Conhecimento, 2008.

4.1.2 *Java Standard Edition*

A plataforma J2SE é a mais popular, pois a mesma é direcionada para o desenvolvimento de aplicações para computadores *desktop*. Assim como as demais plataformas Java, a plataforma J2SE facilita o desenvolvimento das aplicações (SCHAEFER, 2004). Esta plataforma oferece suporte para padrões tais como: *eXtensible*

Markup Language (XML), *Document Object Model* (DOM), *Secure Sockets Layer* (SSL), *Lightweight Directory Access Protocol* (LDAP), e *Common Object Request Broker Architecture* (CORBA).

Outra funcionalidade importante contida na tecnologia J2SE é a tecnologia de *JavaBeans*. Estes são pedaços de código reutilizáveis que podem ser desenvolvidos e montados facilmente, a fim de criar aplicações mais sofisticadas. Basicamente, são blocos de construção das aplicações J2SE. Um *JavaBean* é uma classe Java que segue uma definição especial. Num *JavaBean*, há o conceito de *propriedade*. Uma propriedade, que pode ou não estar associada a uma variável, é equivalente a uma variável interna da classe. Toda propriedade deve possuir dois métodos de acesso, *set()* e *get()*. Um *JavaBean* pode ser referenciado em uma página JSP, separando-se, dessa forma, a interface do programa.

4.1.3 *Java Micro Edition*

A plataforma *Java Micro Edition* ou J2ME, como é mais conhecida, é a especificação Java destinada a DM, tais como telefones celulares, PDA, televisores digitais móveis, sistemas de navegação automotiva, comutadores e roteadores de rede, componentes para automação residencial, entre outros (PALUDO, 2003).

No desenvolvimento de aplicações para DM, há tipicamente dificuldades, tais como variedade de plataformas, ferramentas e incompatibilidade de dispositivos. Diante disso, uma solução muito interessante, e que tem se mostrado uma tendência em ambiente de desenvolvimento para DM, é a adoção da plataforma J2ME. Isso se deve à alta portabilidade das aplicações desenvolvidas nesse ambiente (FIGUEIREDO; NAKAMURA, 2003).

J2ME é um conjunto de especificações, que define uma JVM simplificada, um conjunto de API (Interfaces para Programação de Aplicações) especializadas para pequenos dispositivos, tipicamente DM, e ferramentas direcionadas aos dispositivos com poder de processamento, memória e conectividade menores que um computador *desktop* (PALUDO, 2003).

Conforme Figueiredo e Nakamura (2003) são características do J2ME:

- a) Modularidade e escalabilidade;
- b) Possibilidade de personalização;
- c) Possui um conjunto de tecnologias e ferramentas para o desenvolvimento de aplicações Java para os mais diversos DM;
- d) Facilidade de integração com outras soluções Java, como por exemplo, o J2EE para aplicações corporativas e baseadas na *web*.

A plataforma J2ME contém configurações e bibliotecas especialmente para DM, provendo maior facilidade para o desenvolvedor lidar com as limitações de processamento e memória.

4.1.3.1 Configurações

Devido à grande diversidade de DM e à grande diferença de funcionalidades, sistemas operacionais, capacidade de processamento e memória, e outras, constatou-se a necessidade de uma plataforma com funções básicas, desconsiderando-se detalhes específicos dos dispositivos. Assim, uma configuração de referência possibilita execução e desenvolvimento de aplicações baseadas na especificação J2ME, estabelecendo um conjunto mínimo de funcionalidades.

Uma API de configuração J2ME pode conter algumas API J2SE e outras adicionais. Dispõe-se de duas configurações cujas diferenças essenciais são tipo de conexão e velocidade e capacidade dos dispositivos.

4.1.3.1.1 *Connected Device Configuration (CDC)*

A configuração *Connected Device Configuration (CDC)* especifica um conjunto de tecnologias destinadas à utilização em dispositivos que tipicamente contenham microprocessador/controlador de 32-bits, no mínimo 512 KBytes (Kb) de memória ROM, 256 Kb de memória RAM e algum tipo de conexão persistente. Um exemplo de conexão persistente é a conexão com banco de dados, cuja taxa de transmissão é elevada (MUCHOW, 2004). Utilizando a configuração CDC, pode-se construir interfaces gráficas e aplicações consideráveis computacionalmente em equipamentos que suportem a JVM completa.

4.1.3.1.2 *Connected Limited Device Configuration (CLDC)*

A configuração *Connected Limited Device Configuration (CLDC)* é destinada à dispositivos com recursos de hardware bastante limitados, como processadores de 16 bits e memórias com 512 KB de capacidade. Essa configuração contém uma JVM e um conjunto básico de bibliotecas que possibilita o funcionamento da aplicação Java em dispositivos com tais características (ALECRIM, 2005).

Algumas características de hardware para a configuração CLDC segundo Muchow (2004):

- a) 128Kb de memória para executar o Java;

- b) 32Kb de memória para alocação de memória em tempo de execução;
- c) interface restrita com o usuário;
- d) baixo poder de processamento e consumo de energia, normalmente alimentado por bateria e;
- e) conectividade de rede, normalmente dispositivos sem fio com largura de banda baixa e acesso intermitente;

Segundo Muchow (2004), à medida que a tecnologia oferecer mais capacidade de processamento, memória e recursos, a sobreposição entre essas categorias se tornará maior.

Para tratar dessa ampla variação de recursos e para dar mais flexibilidade à medida que a tecnologia muda foi introduzido o conceito de Perfil.

4.1.3.1 Perfil

Pode-se definir perfil como uma especialização e extensão da API ou JVM, visando melhorar o desempenho e disponibilizar possíveis implementações de funcionalidades e usos específicos para determinados dispositivos. Alguns perfis disponíveis são *Mobile Information Device Profile (MIDP)*, *RMI Profile*, *PDA Profile*, *Personal Java*, dentre outros.

Os perfis do J2ME são, portanto, uma extensão das configurações, definindo bibliotecas para o desenvolvimento de aplicações para um determinado dispositivo. Estas bibliotecas podem ser de interface gráfica, armazenamento persistente, segurança e conectividade. Cada configuração da J2ME pode ter um ou mais perfis associados (SILVA, 2005).

4.1.3.1.1 *Mobile Information Device Profile (MIDP)*

O *Mobile Information Device Profile* (MIDP) é um perfil CLDC que suporta API de interface com o usuário, entrada de dados, manipulação de eventos, persistência de dados, modelo de armazenamento orientado a registro, HTTP, dentre outras características (MUCHOW, 2004). O perfil MIDP pode acessar todas as classes disponíveis na CLDC e quando integrados, o MIDP provê um ambiente de execução Java padrão (SUN, 2007), para DM tais como celulares e PDA.

Conforme Silva (2005) o perfil MIDP é dividido em duas versões. A primeira versão, MIDP 1.0, define todo o conjunto de bibliotecas requeridas pelas aplicações voltadas para DM associados à configuração CLDC. A segunda versão, MIDP 2.0, é uma atualização da especificação MIDP, com recursos novos e refinados. Alguns recursos da especificação MIDP 2.0 são (SILVA, 2005):

- a) Interface com usuário melhorada;
- b) Suporte a funções multimídia e jogos;
- c) Apoio a novos tipos de conexão, como HTTP;
- d) Novas especificações de segurança.

Conforme Schmitt Junior (2004, p. 23), uma aplicação MIDP contém no mínimo uma classe MIDlet (classe que serve para executar o programa no dispositivo ou emulador), podendo em alguns casos conter mais de um MIDlet, sendo chamada de *MIDlet Suite*. A aplicação MIDP, para ser instalada no dispositivo, precisa ser empacotada, criando um *Java Archive* (JAR) que obrigatoriamente contém um arquivo do tipo *manifest*, englobando as informações sobre o MIDlet contido no pacote JAR. Este arquivo contendo as informações sobre a aplicação, configuração e perfil utilizado na mesma, é denominado *Java Application Descriptor* (JAD).

4.2 ACESSO AO BANCO DE DADOS UTILIZANDO JAVA

Um banco de dados é um conjunto de registros dispostos em estrutura regular em que os dados são armazenados para recuperação e reorganização futura. O método ou mecanismo organizado usado para manter e acessar um banco de dados é denominado Sistema Gerenciador de Banco de Dados (SGBD) - *Database Management System* (DBMS). Um sistema gerenciador de banco de dados é uma ferramenta computacional que automatiza a construção e o gerenciamento de um banco de dados a partir de seu modelo.

Normalmente, um SGBD adota um modelo de dados. Um modelo de dados vastamente adotado atualmente é o relacional, cujas estruturas têm forma de tabelas, organizadas por linhas e colunas, onde linhas e colunas podem ser relacionadas a uma ou mais linhas e colunas em outra tabela.

O SGBD realiza a comunicação entre cliente e servidor de banco de dados, podendo o usuário adicionar, excluir, consultar e atualizar registros. O cliente é qualquer aplicação que se comunica com o servidor (estação base), requisitando serviços. O servidor realiza trabalho de gerenciamento de dados, enquanto o cliente é responsável pela interface do usuário e pela própria aplicação, além de administrar uma memória local para solicitação de consultas e armazenamento do resultado das consultas realizadas (ALVES, 2007).

Cientes de banco de dados utilizam *drivers* para mandar comandos SQL ao servidor de banco de dados e para receber resultados, ações e outras respostas do servidor. *Drivers* JDBC são usados por aplicações Java e *applets* para comunicação com o servidor de bancos de dados (BARCELLOS; GARCIA; PELTRINI JÚNIOR, 2004).

Muitos servidores de bancos de dados usam protocolos específicos do fabricante. Isso significa que um cliente de banco de dados terá que aprender uma nova linguagem para conversar com uma base de dados diferente. Entretanto a Microsoft estabeleceu um padrão

comum para a comunicação com bases de dados, chamado "*Open Database Connectivity*" (Conectividade aberta de bases de dados). Antes do ODBC, a maior parte dos clientes de bancos de dados eram específicos para um servidor. Os *drivers* ODBC abstraíram os protocolos específicos de fabricantes, fornecendo interface de programação comum aos Clientes de banco de dados (BARCELLOS; GARCIA; PELTRINI JÚNIOR, 2004).

O JDBC é uma API desenvolvida para aplicações Java. Uma solução pura e eficiente em Java, possibilita conectar aplicativos em rede ou via *Web* a bancos de dados. Isto simplifica o acesso à base de dados para os usuários conectados em rede ou via *Web* (BARCELLOS; GARCIA; PELTRINI JÚNIOR, 2004).

4.3 REDES DE COMPUTADORES

Redes de computadores são constituídas por módulos que compartilham informações e recursos por meio de um sistema de comunicação (SOARES; LEMOS; COLCHER, 1995). Utilizam-se de meios de transmissão que se diferenciam por características tais como banda passante, tipo de conexão, limitação de abrangência geográfica, custo, confiabilidade, segurança, dentre outros. Os meios físicos são divididos em: guiados, que envolvem cabos e fibras óticas e; não-guiados, que envolvem comunicação por ondas eletromagnéticas (SOARES; LEMOS; COLCHER, 1995). Dentre estes últimos estão as redes sem fio, cuja comunicação ocorre através de ondas eletromagnéticas por meio de sinais de radiofrequência, dispensando totalmente a utilização de cabos.

Diversas tecnologias estão incluídas nas redes sem fio, mas as de interesse nesta trabalho estão relacionadas com o Padrão 802.11. As redes sem fio são atualmente vastamente utilizadas nos mais variados ambientes, provendo flexibilidade, mobilidade e a tendência eliminação de cabos e redução de gastos (DUARTE, 2003).

A mobilidade é uma vantagem que possibilita que dispositivos possam ser movidos ao longo da rede sem que a conexão entre os elementos da rede seja perdida. (RUFINO, 2005).

Para o acesso ao banco servidor pela aplicação, localizada no DM, usa-se uma rede sem fio, denominada *wi-fi*, baseada no padrão IEEE 802.11. Nesta categoria de redes, há vários tipos de redes que são: Redes Locais Sem Fio ou *Wireless Local Area Network* (WLAN), Redes Metropolitanas sem Fio ou WMAN (*Wireless Metropolitan Area Network*), Redes de Longa Distância sem Fio ou *Wireless Wide Area Network* (WWAN), redes *Wireless Local Loop* (WLL) e o novo conceito de Redes Pessoais Sem Fio ou *Wireless Personal Area Network* (WPAN).

4.3.1 Wi-Fi

Pode-se dizer que a tecnologia *Wireless* viabiliza a mobilidade e a praticidade da informação independente do lugar, necessária aos DM e demais dispositivos portáteis. Redes *wireless* possibilitam acesso aos dados onde e quando for necessário (SCHAEFER, 2004).

As WLANs têm sido usadas em campus de instituições de ensino, prédios comerciais, *resorts*, aeroportos, condomínios residenciais, medicina móvel no atendimento aos pacientes, transações comerciais e bancárias. Além disso, as WLANs também são empregadas onde não é possível atravessar cabos, como por exemplo, em construções antigas ou tombadas pelo patrimônio histórico (BONILHA, 2003).

Atualmente, diversas operações são executadas por intermédio das redes *wireless*, tais como operações bancárias, compras *on-line*, dentre outras. Segurança e privacidade atualmente requisitos imprescindíveis.

Além das estratégias de segurança utilizadas nas redes cabeadas, para as redes sem fio foram requeridas recentes mecanismos (TANENBAUM, 1997). Conforme Albergaria et al (2006) o padrão IEEE 802.11 fornece serviços de segurança dos dados por meio de dois métodos: autenticação e criptografia.

4.3.1.1 Endereçamento MAC

Os dispositivos de comunicação atualmente possuem um número único de identificação, controlado pelo *Institute of Electrical and Electronics Engineers* (IEEE), definido pelo fabricante. Trata-se de uma informação para restrição de acesso a uma rede. Isto possibilita que apenas dispositivos autorizados possam acessá-la. Entretanto, há atualmente *softwares* que emulam este código.

4.3.1.2 *Wired Equivalent Privacy (WEP)*

Para prover privacidade na comunicação entre dispositivos em rede *wireless*, o Padrão 802.11 possibilita que os dados sejam criptografados. O protocolo WEP está presente em todos os dispositivos do Padrão 802.11. As informações trafegadas são codificadas pelo algoritmo simétrico RC4 que usa criptografia de 64 e 128 *bits* fazendo uso de chaves secretas compartilhadas entre o cliente e servidor. A autenticação é realizada na camada de enlace. Utiliza-se adicionalmente o padrão CRC-32 para checagem de redundância cíclica nos pacotes de dados verificando a integridade destes. Autenticação é o serviço que verifica se um usuário está autorizado a usufruir de acesso à rede ou se um computador pode se comunicar com outro em região de cobertura distinta.

O protocolo WEP requer baixa complexidade computacional na codificação das informações (RUFINO, 2005).

4.3.1.3 *Wi-fi Protected Access (WPA)*

O protocolo WPA é uma evolução do protocolo WEP. Suas principais diferenças estão na codificação dos dados e na autenticação de usuários que faz utilização do padrão 802.1x e *Extensible Authentication Protocol (EAP)* (RUFINO, 2005).

4.3.1.4 *Certificação Digital*

Certificação Digital é um importante método de segurança que consiste em uma estrutura que relaciona dados de um usuário, a um par de chaves assimétricas que, usadas em conjunto, fornecem a identificação do titular. A estrutura referida é chamada de certificado digital (GOULARTE, 2006).

Um certificado digital é um documento eletrônico assinado digitalmente e cumpre a função de associar uma pessoa ou entidade a uma chave pública. As informações públicas contidas num certificado digital são o que possibilita colocá-lo em repositórios públicos (BRASIL, 2007).

O padrão X.509 estabelece que um certificado digital deve conter, entre outras, as seguintes informações: Versão, número de série, período de validade, emissor, usuário, chave pública do usuário e a assinatura digital do emissor. A veracidade dos dados contidos no certificado digital é assegurada pela entidade que o emite, a chamada Autoridade Certificadora. A aceitabilidade do certificado dependerá da confiança dos usuários nas práticas de trabalho da autoridade Certificadora. A Autoridade de Registro tem o papel de

verificar a identidade do usuário solicitante do certificado, de forma presencial e atendendo a padrões estabelecidos pela Autoridade Certificadora a qual está vinculada (VASCONCELLOS LUZ, 2004).

4.4 *FRAMEWORKS*

Um *framework* pode ser considerado como uma infra-estrutura de classes que oferecem o comportamento necessário para implementar aplicações dentro de um domínio específico, funcionando com um molde para as aplicações. É um projeto abstrato para um tipo de aplicação (GAMMA et al, 1995 apud COSTA, 2001). Um *framework* orientado a objeto é diferente de um pacote de classes ou de uma API, pois indica o modo como as classes devem ser instanciadas para colaborar na construção de uma aplicação (CARVALHO, 2005).

Os *frameworks* têm como objetivo facilitar para o engenheiro de software a construção de aplicações com a menor quantidade de código escrito possível, além de propiciar que boas soluções possam ser reutilizadas dentro de um mesmo domínio de aplicações. Eles são construídos para serem flexíveis e possibilitar a inclusão de novas funcionalidades (CARVALHO, 2005).

No desenvolvimento do S-RES foram utilizados dois *frameworks*, o *Hibernate*, usado no mapeamento de informações em formato XML para consultas ao banco de dados, e o *SpringMVC*, utilizado na construção da aplicação.

4.4.1 *Hibernate*

O *Hibernate* é uma ferramenta para mapeamento objeto/relacional para programas em Java. O termo mapeamento objeto/relacional (ORM) se refere à técnica de mapeamento

em representação de dados de uma modelo objeto para um modelo relacional de dados baseado em SQL (HIBERNATE, 2007).

O *Hibernate* não é usado apenas para tomar conta do mapeamento de classes Java para tabelas do banco de dados (e de tipos de dados Java para tipos de dados SQL), como também fornece os dados da consulta com facilidade, obtendo as estruturas de recuperação e podendo reduzir significativamente o tempo gasto com outra mudança manual de dados em SQL e JDBC (HIBERNATE, 2007).

4.4.2 *SpringMVC*

O *Spring* é um *framework* popular de código-fonte que pode facilitar o desenvolvimento de aplicativos em J2EE. Ele consiste em um contêiner que serve para gerenciar componentes, e um conjunto de serviços para interfaces de usuário, transações e persistência da *Web* (NETBEANS, 2008).

O *framework Spring* acrescenta um apoio significativo quando se utiliza o mapeamento O/R das camadas de sua escolha na criação de aplicações de acesso aos dados. Primeiro de tudo, você deve saber que quando você começou a usar o apoio do *framework* para mapeamentos O/R, você não tem que ir até ao fim do código (JOHNSON, 2004).

O *framework* tem um conjunto de componentes que contem os relacionamentos dos objetos criados. O *Spring Framework* utiliza um arquivo em formato XML para especificar o relacionamento entre os componentes e os valores para outras propriedades que precisem ser configuradas.

5 TRABALHOS CORRELATOS

Em COSTA (2001), são avaliados metodologias de desenvolvimento de S-RES e implementação de seus conceitos num sistema baseado na *Web*. Também são abordados definição do problema, avaliação da realidade do mercado em relação ao desenvolvimento e adequação de um S-RES.

É feita referência ao Registro Médico Orientado ao Problema, sua forma de análise de dados e os problemas que enfrentava o autor em SAVAGE (2001). Também é explicado o modelo de PMOR, suas vantagens e a descrição do problema encontrado e posteriormente resolvido por Savage.

É relatado em Murakami (2003) a metodologia para elaboração de um Registro Eletrônico para DM. É descrita desde a criação de uma rede sem fio até o S-RES totalmente implementado.

Em GOULARTE (2006) é proposta a modelagem de um sistema de informação médica para as clínicas de saúde integradas da UNESC, com base em registros eletrônicos e de padrões na área da saúde. A pesquisa aborda aspectos técnicos e teóricos relacionados ao Prontuário do Paciente, sendo ele em papel ou eletrônico, segurança no transporte de informações, aspectos legais que envolvem o registro médico e a importância que a Engenharia de Software representa no processo de desenvolvimento de um sistema. O projeto do sistema elaborado nesse trabalho seguiu a metodologia do desenvolvimento do sistema PDS-DATASUS.

Em MARTHA et. al. (2006) é proposto o desenvolvimento, por meio da computação móvel, de um Registro Eletrônico para ambiente *web*. É apresentado um histórico sobre o S-RES, cuja comunicação com os DM se dá via *Web*.

Em STRACHOSKI (2006) é proposta uma modelagem conceitual de um prontuário eletrônico com base em normas de construção de S-RES e padronização da informação médica. São apresentados os aspectos teóricos e técnicos referentes aos prontuários em papel e S-RES. São discutidos padrões e normas para desenvolvimento S-RES envolvendo uma diversidade de fontes e termos usados na área. É discutido no trabalho a importância da Engenharia de Software e dos seus paradigmas na construção do S-RES.

Em FRASSON JÚNIOR (2008), são abordadas vulnerabilidades no acesso a um S-RES nos itens que compõem o ambiente associado a uma aplicação móvel, visando propor recomendações para orientar a escolha das tecnologias adequadas. Para validação das recomendações, foi realizado um estudo de caso na aplicação móvel de acesso a um S-RES. Foram identificadas vulnerabilidades e avaliados métodos para minimizá-las.

Em MEZARROBA ET AL (2008), é descrita a utilização das tecnologias e as considerações para o desenvolvimento de S-RES específico para Unidades de Terapia Intensiva. O desenvolvimento segue as orientações do Manual de Requisitos de Segurança, Conteúdo e Funcionalidades para Sistemas de S-RES (SBIS; CFM, 2009). Utilizaram-se tecnologias multiplataforma e softwares livre para o desenvolvimento das aplicações. O S-RES resultante é multiplataforma e implementa segurança de informações mediante suporte à assinatura digital. São discutidos e utilizados *Frameworks*, padrões arquiteturais e padrões de projeto almejando a expansibilidade do *software*.

6 TRABALHO DESENVOLVIDO

O presente trabalho descreve o desenvolvimento de um protótipo de uma aplicação para o acesso a informações do prontuário entre um servidor remoto e um cliente utilizando um DM por meio de uma rede wireless.

Nesse capítulo, será discutido o desenvolvimento do projeto, detalhando as etapas realizadas para a construção do sistema e os recursos utilizados.

A metodologia dessa pesquisa baseou-se nas seguintes etapas: análise de conteúdo e funcionalidades do mUTInfo 2.0, problemas encontrados, implementação, resultados obtidos e projetos futuros para uma expansão do sistema. Também serão abordadas algumas considerações sobre ética em relação aos dados acessados S-RES.

6.1 COMUNICAÇÃO PRONTUÁRIO - DM

Atualmente sabe-se que o profissional da área da saúde não pode ficar restrito a um programa de computador somente para relatar os dados de uma consulta e atualizar os dados do paciente. Os dados médicos devem estar disponíveis a qualquer hora e local. Em situações específicas, tais como em emergências, por exemplo, o profissional pode acessar informações sobre determinado paciente, agilizando o atendimento (MALUTA, 2001).

Com isso, é desejável que o S-RES esteja sempre disponível e acessível. O desafio atual do S-RES é disponibilizar as informações que ele possui para o profissional de saúde em qualquer lugar que ele necessite e a qualquer hora.

No caso de um hospital, a infra-estrutura de rede de acesso desse sistema pode ser disponibilizada através de uma tecnologia de conectividade sem fio. DM sem fio podem se

conectar a essa rede, podendo ser facilmente transportado pelo médico. Cada um desses DM pode ser dotado de um sistema de acesso ao S-RES que possibilite acesso à base de dados. Nesse tipo de infra-estrutura, os dispositivos móveis deverão interagir com um banco de dados instalado em um servidor de banco de dados. No que diz respeito a essa interação, um primeiro passo é a procurar a melhor forma dos dispositivos móveis interagirem com uma grade de computadores utilizando os recursos oferecidos por esta, com o objetivo de complementar as limitações inerentes aos DM tais como capacidades de processamento, memória, armazenamento e comunicação. Inúmeras dificuldades estão presentes nessa integração, pertencentes à utilização desses dispositivos em rede como as instabilidades de enlace de dados, interoperabilidade e, sobretudo, segurança.

A computação móvel requer uma estrutura avançada que integre protocolos de suporte, mecanismos e ferramentas adequadas. Ambientes abertos devem superar a heterogeneidade do sistema e garantir um nível de segurança apropriado. Soluções devem visar garantir a interoperabilidade entre sistemas diferentes e componentes legados (MALUTA, 2001).

A utilização de unidade sem-fio amplia imensamente o horizonte de unidades que podem estar envolvidas, envolvendo desde instrumentação eletrônica por meio de sensores até a monitoração de resultados por meio de DM. Neste caso, tanto as redes fixas quanto as redes sem fio necessitam de técnicas de comunicação e gerenciamento específicas que atendam aos requisitos dessa arquitetura, atentando-se para algumas limitações típicas aos dispositivos móveis, como desempenho de CPU, armazenamento secundário, economia de energia e largura de banda disponível (BRASIL, 2006).

A conexão e a comunicação do DM com o banco de dados (BD) do S-RES acontece por meio da rede *wireless* para envio e recebimento das informações necessárias.

O S-RES também utilizada conexão com o BD via rede cabeada, mas neste caso para computadores tipo desktop.

Na primeira tela, de *login*, o S-RES se conecta ao banco para a verificação de usuário e senha. Se os dados estiverem corretos, é passado para a utilização das funcionalidades do S-RES.

Após essa etapa, acontece o carregamento de dados dos pacientes do banco de dados. Isso é feito pela biblioteca kxml, que é um interpretador de XML, com proporções reduzidas. Ele é utilizado com o J2ME. Nessa interpretação de código XML, é usado o *framework Hibernate* para a comunicação com o banco de dados, mapeando os dados contidos em formato XML, tornando as consultas mais simples e rápidas.

Junto com o *Hibernate*, é utilizado um *framework* para o desenvolvimento de uma aplicação *web* para a importação dos dados contidos no banco. Esse *framework* é o *Spring*.

Ele provê uma integração com o *Hibernate* em termos de gestão dos recursos, apoio à implementação DAO, operações e estratégias. Para o *Hibernate*, a primeira classe tem suporte com os lotes apoio de características do IoC, abordando muitas questões típicas ligadas à integração com o *Hibernate*. Todos estes pacotes de apoio para mapeadores O/R (Objeto Relacional) contemplam a transação genérica *Spring* e hierarquias DAO *exception*. Normalmente existem dois estilos integração: ou utilizando "*templates*" DAO *Spring* ou codificação DAO para API *Hibernate*. Em ambos os casos, o DAO pode ser configurado através de dependências dos objetos e participar na gestão da operação e dos recursos *Spring*. (JOHNSON, 2004).

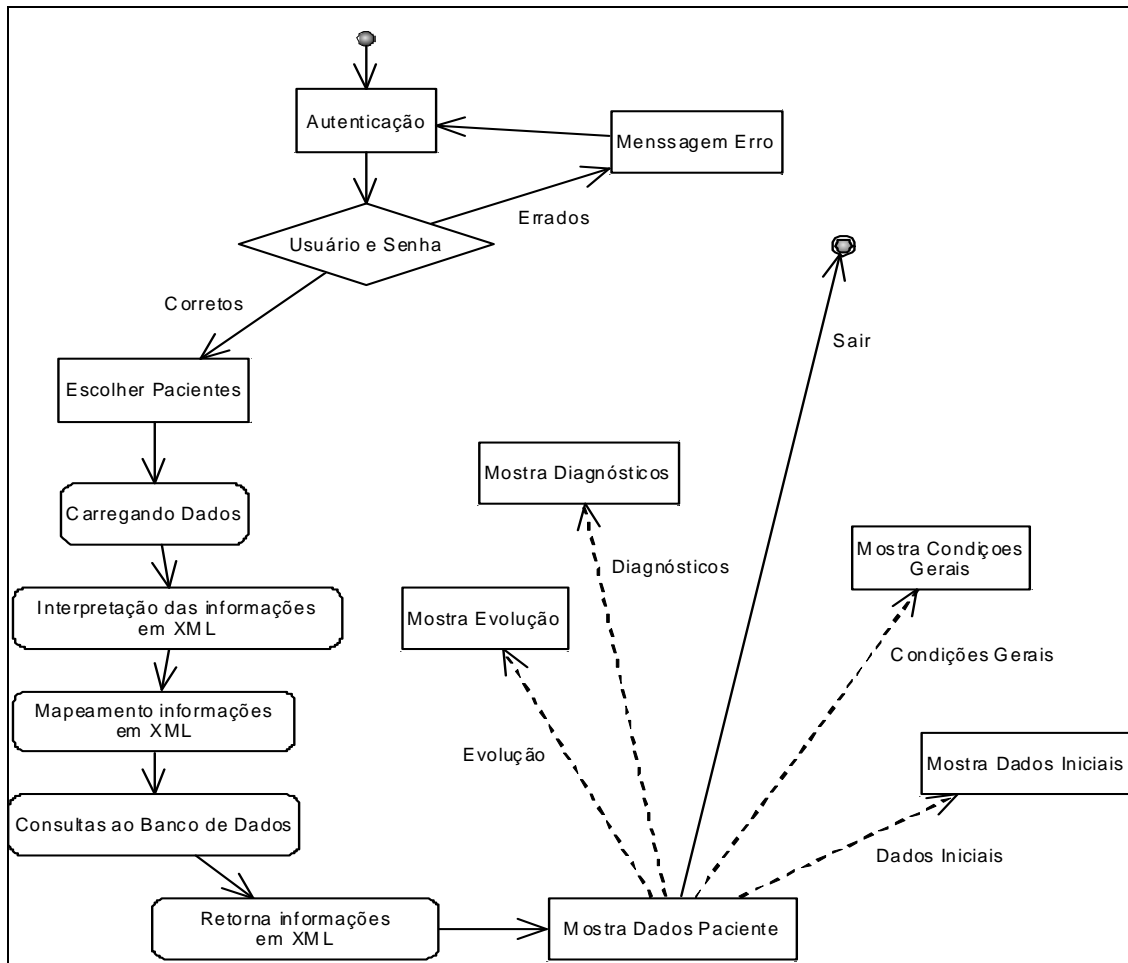


Figura 9: Fluxograma da comunicação do S-RES com o Banco de Dados
 Fonte: KIRON (2009).

6.2 O REGISTRO ELETRÔNICO DE PACIENTE - mUTInfo 2.0

Recentemente, foi desenvolvido um sistema de Sistema de Registro Eletrônico de Saúde (S-RES), baseado na tecnologia Java e banco de dados *PostGresSQL*, denominado UTInfo 2.0.

O software tem funcionalidades de cadastro de médicos e pacientes. Ele tem coletado em seu banco de dados as informações dos pacientes, que passaram pela UTI, como: evolução médica, doenças contraídas, medicamentos utilizados, informações cadastrais, entre outros. Nele também se encontram informações sobre a ocupação de leitos, e entre outras funcionalidades referentes à ocupação da UTI do hospital.

No entanto, esse sistema requer que os dados de pacientes que passam pela UTI, bem como as evoluções médicas, estejam disponíveis a qualquer hora, em determinados setores do hospital, não se limitando o acesso às informações em somente um computador.

Diante dessa necessidade, o Grupo de Pesquisa em Informática Médica e Telemedicina (Kiron), por meio deste trabalho, vêm desenvolvendo um Sistema de Registro Eletrônico de Saúde para Dispositivos Móveis, denominado mUTInfo 2.0.

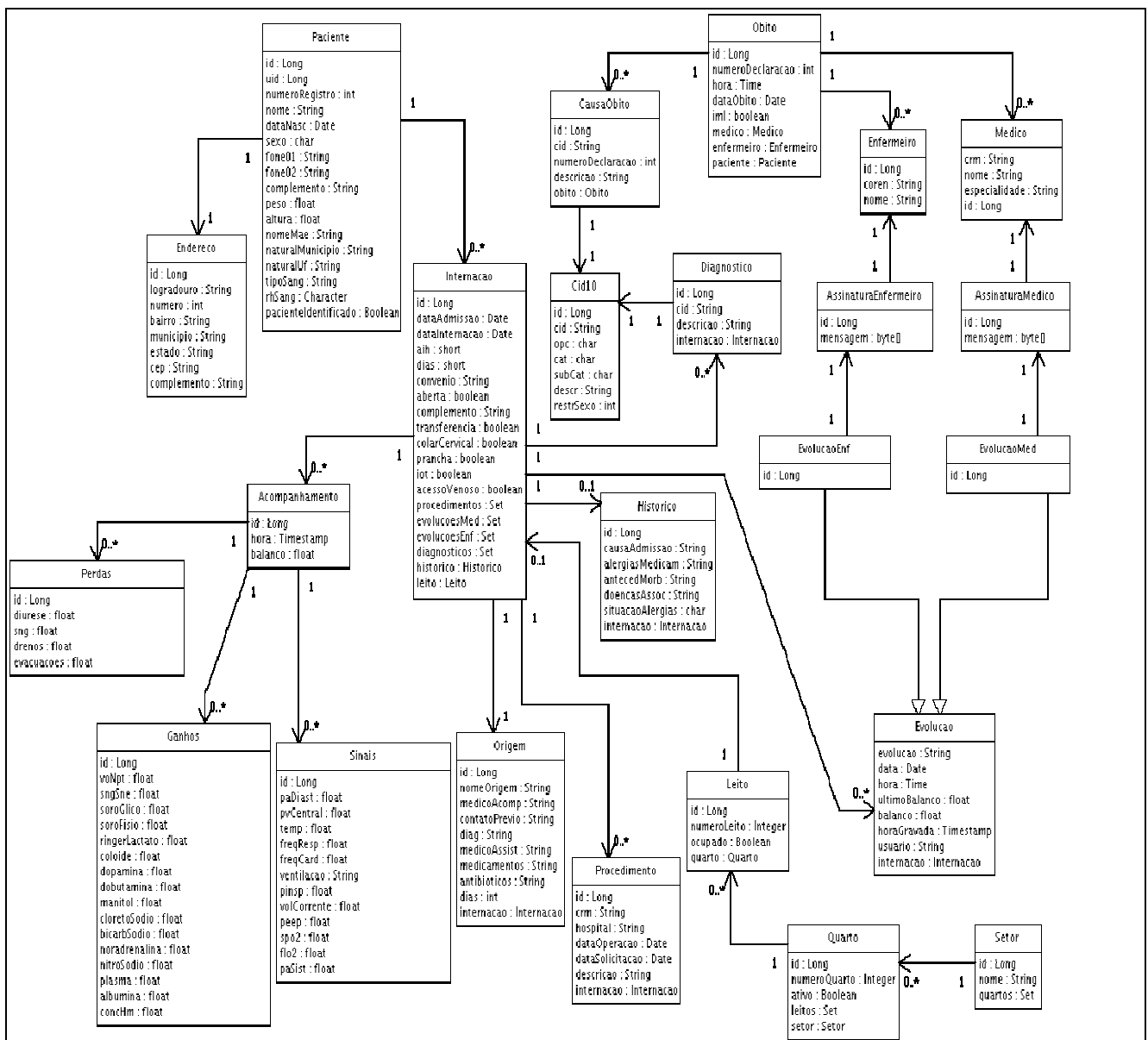


Figura 10: Diagrama de Classe Banco de Dados UTInfo 2.0
Fonte: KIRON (2009).

Esta versão do mUTInfo 2.0 foi implementada no ambiente de programação J2ME, por meio da IDE *NetBeans* 6.1, integrada ao banco de dados *PostgreSQL*. As informações contidas no banco, em formato XML, foram exportadas com ajuda de uma biblioteca chamada *kxml*.

A integração das funcionalidades do sistema se dá por meio de interfaces gráficas, onde a partir de um paciente é possível obter todos os dados do mesmo. Para essa versão, optou-se pelas funcionalidades associadas às informações por meio de uma interface simplificada. Uma interface gráfica mais atrativa necessitaria de uma biblioteca específica aliada a renderização gráfica que acarretaria em um maior consumo de memória e complexidade computacional, que são reduzidas nestes dispositivos.

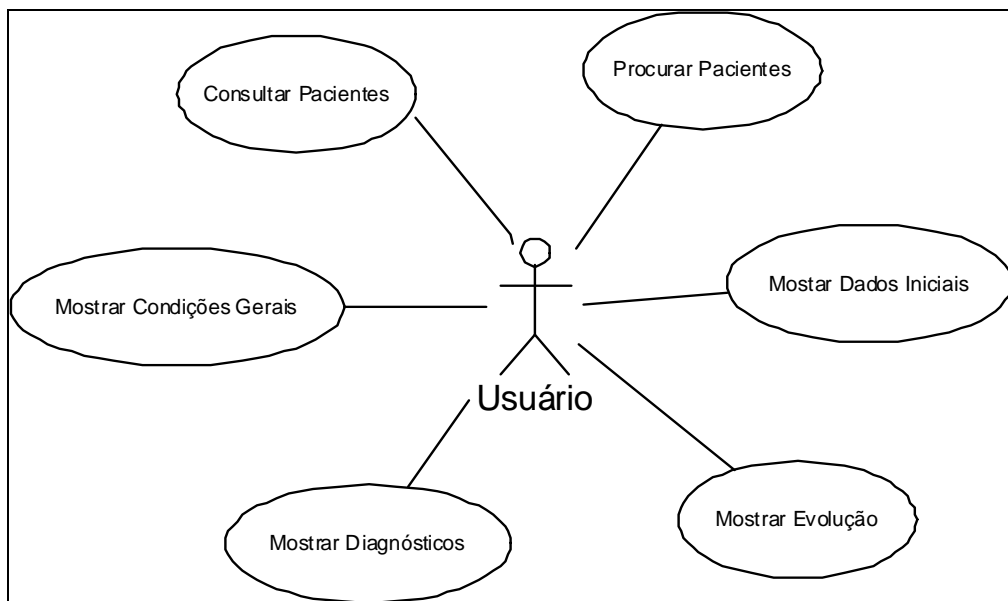


Figura 11: Diagrama de Caso de Uso das funcionalidades do mUTInfo 2.0
Fonte: KIRON (2009).

O prontuário acessa o banco por meio da rede *wireless*. A primeira tela da aplicação é a tela de *login*, onde o usuário faz a autenticação para se conectar ao banco de dados. O *login* e a senha podem ser um conjunto de caracteres alfanuméricos, sem restrições para símbolos. Um método seguro para desenvolver a autenticação e transmissão de dados é o da criptografia.

O método de criptografia é conhecido como WEP (*Wireless Equivalent Privacy*). Ela codifica as entradas e saídas de dados para que somente os equipamentos que estão em comunicação entendam o que se está sendo transmitido no momento. Esse é um método seguro de comunicação feitas por redes sem fio entre os diversos equipamentos. Um dos métodos mais utilizados atualmente consiste em duas chaves: publica e privada.

A chave publica serve para “embaralhar” os dados, de forma que as informações contidas em um arquivo não sejam descobertas. Já a chave privada serve para identificar os dados embaralhados, que somente o destinatário conhece.

No campo senha, é colocada uma mascara para evitar que usuários possam descobrir a senha de outros (Figura 12).

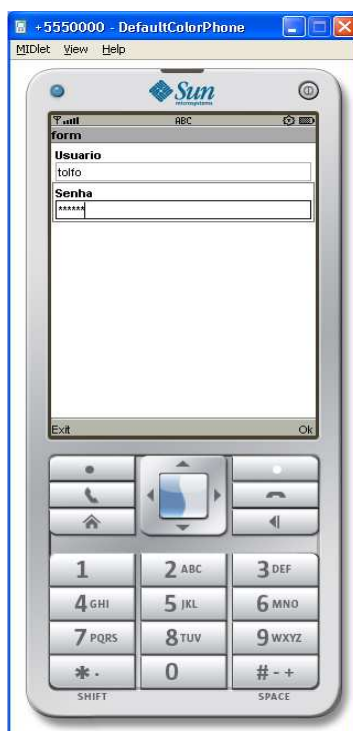


Figura 12: Tela de Login mUTInfo 2.0
Fonte: KIRON (2009).

Para o carregamento no banco, tem-se uma barra de *status*, mostrando a evolução da conexão via rede HTTPS com o banco de dados. Essa evolução leva em conta todos os passos que vão desde o envio dos dados, até a validação de nome e senha.



Figura 13: Tela de *Status*
Fonte: KIRON (2009).

Na aplicação também existe uma tela para controle de erros. É implementada uma tela de alerta, que é apresentada quando existe algum erro com a conexão do banco de dados ou com o servidor.



Figura 14: Tela de alerta de erro
Fonte KIRON (2009).

Na tela apresentada, ocorre um erro com a falta do certificado. O certificado serve como base de segurança para uma conexão segura com o servidor. A tela apresenta o erro e fica 5 segundos, após esse tempo, volta-se a tela que estava sendo utilizada anteriormente ao erro.

Após a conexão com o servidor concluída, passa-se a tela para procura de pacientes. Nessa tela, pode-se procurar pacientes por nome e situação de internação (Figura 13). Pode-se ter internação aberta ou fechada e os internados podem ter óbito ou alta. A situação de internação vem ao caso de o leito ainda estar sendo ocupado pelo paciente.

Nas figuras que mostram pesquisas e informações de pacientes, é colocado os nomes sombreados. As informações utilizadas nesse projeto são de uma Base de Dados real, assim não sendo possível mostrar as informações contidas, com a possibilidade de sofrer punições previstas em lei, segundo o Código Civil, Código Penal e Código do Processo Penal.



Figura 15: Tela Procura Pacientes
Fonte: KIRON (2009).

Na Figura 16 são mostrados os pacientes encontrados na pesquisa. Os pacientes são procurados pelo nome e carregados. Serão procurados pacientes em função da letra que é contida no nome ou pelo nome completo. É criado um vetor com os pacientes encontrados e colocados os num *choice group* para escolha.



Figura 16: Tela Carrega Pacientes
Fonte: KIRON (2009).



Figura 17: Carregar Dados Gerais
Fonte: KIRON (2009).

Após a escolha do paciente, terá comando para acessar os dados principais do paciente. Os dados carregados são apresentados na tela principal do paciente, conforme mostrado na Figura 18.



Figura 18: Tela de Dados Gerais do Paciente
Fonte: KIRON (2009).

Os dados do paciente selecionado são carregados no banco apenas uma vez, ao caso que nas demais telas já foram inseridas as informações necessárias no primeiro acesso ao banco de dados. Para povoamento dos formulários com as informações, é utilizada uma função de *update*. Essa função pega cada dado do paciente e mostrado em uma *string*. Ela é acessada para cada formulário especificamente, pois a aplicação precisa capturar as características de cada formulário e acrescentar as informações e funcionalidades necessárias. Nas Figuras 19, 20, 21 e 22, são apresentadas as telas de dados iniciais, diagnósticos de entrada, condições gerais e evolução médica, respectivamente, com as informações capturadas no banco de dados, já devidamente carregadas e povoadas.

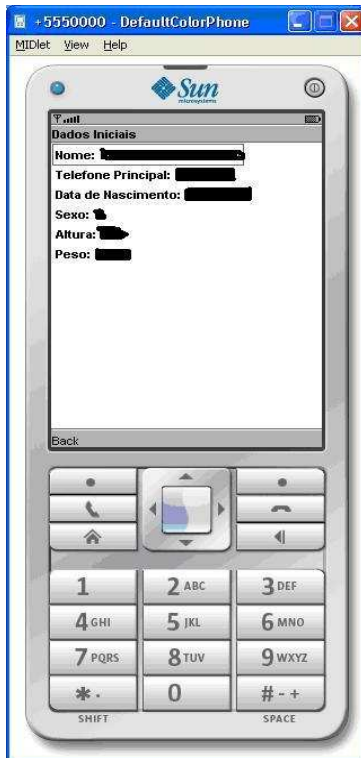


Figura 19: Tela de Dados iniciais
Fonte: KIRON (2009).

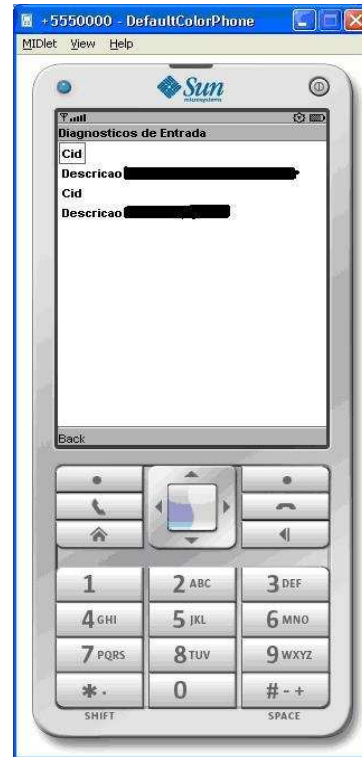


Figura 20: Tela de Diagnósticos de Entrada
Fonte: KIRON (2009).



Figura 21: Tela de Condições Gerais
Fonte: KIRON (2009).



Figura 22: Tela de Evolução Médica
Fonte: KIRON (2009).

Nas figuras apresentadas são mostrados os dados recebidos da consulta feita no banco. Essas informações foram carregadas em formato XML e mostradas no dispositivo móvel.

A seguir, são mostrados os resultados obtidos nessa pesquisa.

6.3 RESULTADOS OBTIDOS

Após a conclusão das etapas de implementação da aplicação móvel, foram feitas uma seqüência de testes. Os testes foram feitos usando um computador com processador Athlon XP a 1.7 GHz e com 1 GB de memória RAM. No computador foi instalado um emulador para J2ME, o WTK⁴. O emulador do dispositivo móvel usado foi também executado nesta máquina (já instalado com a IDE utilizada).

No caso do DM, foi usado um Nokia E.61 com 32Mb de memória e com a JVM da Sun instalada. Neste caso não existe emulação. A MIDlet foi executada no próprio dispositivo, através de *download* da aplicação já compilada e em nível de execução. A ligação do PDA com o computador onde encontrava-se o servidor foi feita através de comunicação via rede sem fio.

Os dados contidos no banco foram acessados com sucesso, sendo apresentados nas telas de informações do paciente. O servidor comportou-se da forma esperada, conectando-se ao dispositivo móvel. Alguns problemas foram encontrados no andamento da implementação, os quais estão detalhados a seguir, junto com as soluções encontradas.

⁴ *Wireless Toolkit* – O *toolkit* de desenvolvimento de aplicações MIDP da Sun.

6.4 PROBLEMAS ENCONTRADOS

Constatou-se que o dispositivo móvel apresentou travamento quando havia grande número de registros encontrados. Isso é devido à reduzida memória dos DM. Esse problema foi solucionado em parte por um refinamento das consultas no banco de dados, possibilitando um número menor de registros encontrados e mostrados em tela, consumindo menos memória do DM.

Outro problema foi encontrado no redirecionamento para o banco feita em redes externas. O servidor do BD, localizado no Laboratório de Informática Aplicada – Kiron - não conseguia se conectar ao dispositivo móvel. Quando se requisitava a conexão, o certificado digital não conseguia enviar os dados criptografados, ou apresentava assinatura digital inválida. O problema é originário de *bug* do Sistema Operacional Windows, que não achava o local de origem do certificado. A solução encontrada foi um pequeno código de execução que repassava a leitura do certificado sendo como *home* (local) e assim, dando o local de origem do certificado.

```
package net.unesc.kiron;

public class HomeDir {

    public static void main(String argv[]) {
        System.out.println(System.getProperty("user.home"));
    }
}
```

Figura 23: Classe HomeDir
Fonte: KIRON, 2008

CONCLUSÃO

Profissionais de diversas áreas atualmente necessitam de mobilidade e acesso a informações, enviando e recebendo notícias, documentos e diversos tipos de informação a qualquer hora e de forma segura. O profissional da saúde também requer acesso a informações de um prontuário independentemente de restrições de tempo e espaço. A mobilidade e acesso constante a essas informações são fundamentais. Nesse contexto, o desenvolvimento de um Sistema de Registro Eletrônico em Saúde para Dispositivos Móveis atende essas condições.

Foram analisados requisitos necessários e disponíveis dos dispositivos móveis, tecnologias para implementação do aplicativo, tecnologias para envio de informações pela rede, de banco de dados e o S-RES foi implementado com sucesso.

Um aspecto positivo é que o projeto desenvolvido pode colaborar na agilidade no atendimento a pacientes.

Conclui-se também que o sistema precisa ser melhorado em suas opções de carregamento no servidor, segurança e consumo de memória, com a finalidade de melhorar seu desempenho.

Em relação a trabalhos futuros, sugere-se a evolução do S-RES, melhorando sua interface gráfica, adicionando mais funcionalidades para os médicos e plantonistas, a partir de análise de requisitos e funcionalidades, e customizações para outras plataformas e DM.

REFERÊNCIAS

- ALBERGARIA, J. O. **Cidades Digitais: O projeto Campo Digital**. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialcdigital/default.asp> Acessado em: 28 out 2008.
- ALECRIM, E. **JSE, JEE e JME: uma breve explicação**. Disponível em: <http://www.infowester.com/versoesjava.php>. Fev. 2005.
- ALVES, R. A. **Um Estudo sobre Segurança em Banco de Dados Móveis**. Recife: Universidade Federal de Pernambuco, 2007.
- ARAÚJO, F. B. **DESENVOLVIMENTO DE SOFTWARE PARA DISPOSITIVOS MÓVEIS**. Centro de Estudos em Telecomunicações – CETUC. Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2006. Disponível em: www.puc-rio.br/pibic/relatorio_resumo2006/relatorio/CTC/Cetuc/F%E1bio%20Bicalho%20de%20Araujo.pdf
- ARMSTRONG, E. et al. **The J2EE 1.4 Tutorial**. Documento Técnico. [On-Line] Disponível em: <http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html> Dezembro, 2005.
- BARCELLOS, H. A.; GARCIA, F. C.; PELTRINI JÚNIOR, J. **JDBC**. Pontifícia Universidade Católica do Rio Grande do Sul, 2004.
- BERNAL, P. S. M.. **Comunicações Móveis: Tecnologias e aplicações**. São Paulo: ed. Érica, 2002.
- BLAIR, J. S. An Overview of Healthcare Information Standards. IBM Corporation White Papers. Disponível na Internet. Acesso em 13/Mar/2009. <http://www.hipaonet.com/cpri.htm>.
- BONILHA, C. **Wireless Lan (WLAN)**. Artigo. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialwlan/default.asp>
- BRASIL. **Projeto MACC - Medicina Assistida por Computação Científica**. Laboratório Nacional de Computação Científica – LNCC / MCT, Abril, 2006.
- CARDOSO, J.; ROCHA, A.; LOPES, J. C. **M-GIS – Sistema Móvel Interoperável de Informação Geográfica**. Faculdade de Engenharia da Universidade do Porto, 2003.
- CARVALHO, W. V. **Um Ambiente de Desenvolvimento de Aplicações Multi-Plataformas e Adaptativas para Dispositivos Móveis**. [Dissertação] Fortaleza (CE): Departamento de Computação da Universidade Federal do Ceará; 2001.
- CBCD. “CID-10 - Classificação Estatística Internacional de Doenças e Problemas Relacionados à Saúde”, 2008. Online Disponível na Internet <<http://www.datasus.gov.br/cid10/v2008/cid10.htm>> acessado em 15 maio 2009.

CFM, In: Conselho Federal de Medicina. Online Disponível na Internet
<http://www.portalmedico.org.br/novoportal/index5.asp>, acessado em 01/05/2009.

COSTA, C. G. A. **Desenvolvimento e Avaliação Tecnológica de um Sistema de Prontuário Eletrônico do Paciente, Baseado nos Paradigmas da World Wide Web e da Engenharia de Software.** [Dissertação] Campinas (SP): Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas; 2001.

CRISPIM JUNIOR, C. F.; FERNANDES, A. M. da R.; **Análise das Aplicações Móveis Existentes na Área da Saúde.** In: X Congresso Brasileiro de Informática em Saúde, Florianópolis, 2006.

DEPINÉ, F. M. **Protótipo de software para dispositivos móveis utilizando Java ME para cálculo de regularidade em rally.** 2002. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2002.

DORILEO, É. A. G. et al. **Estruturação da Evolução Clínica para o Prontuário Eletrônico do Paciente.** In: X Congresso Brasileiro de Informática em Saúde, Florianópolis, 2006.

D'ORNELLAS, M. C.; ROCHA, R. P. **Acesso e Privacidade: em busca da segurança das informações em bancos de dados médicos.** In: CBIS 2002, VIII Congresso Brasileiro de Informática em Saúde. Natal, 2002.

DUARTE, L. O. **Análise de Vulnerabilidades e Ataques Inerentes a Redes Sem Fio 802.11x.** 2003. 66 f. Trabalho de Conclusão de Curso (Bacharel) - Curso de Ciências de Computação, Departamento de Ciências de Computação e Estatística do Instituto de Biociências, Letras e Ciências Exatas, Centro Universitário de João Pessoa, São José do Rio Preto, 2003.

EVANS, E. **Domain-Driven Design: Tackling Complexity in the Heart of Software.** Boston: Addison-Wesley; 2004.

FIGUEIREDO, C. M. S.; NAKAMURA, E. **Computação Móvel: Novas Oportunidades e Novos Desafios.** T&C Amazônia, ano 1, n.º 2, jun. 2003.

FIGUEIREDO, T. H. de P.. **MultiMAD: Uma ferramenta multimodelo de desenvolvimento de aplicações para dispositivos móveis.** 2005. Dissertação (Pós Graduação) - Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, 2005.

FRANÇA, T.; NARDON, F. B.; NAVES, H. **Construção de Aplicações em Saúde Baseadas em Arquétipos.** In: XI Congresso Brasileiro de Informática em Saúde, 2008, Campos do Jordão. XI Congresso Brasileiro de Informática em Saúde, 2008.

FRASSON JÚNIOR, D. **Segurança no desenvolvimento da Aplicação Móvel do REP UTInfo 2.0: Análise de Vulnerabilidade em Comunicação de Dados entre Dispositivos Móveis e Servidor.** 2008. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade do Extremo Sul Catarinense.

GAMMA, E. et al. *Design Patterns*. Addison-Wesley, 1995.

GOULARTE, F. B. **Projeto de um Sistema de Prontuário Eletrônico Baseado na Web para as Clínicas Integradas da Saúde**. Criciúma: UNESC, 2006.

HIBERNATE. **HIBERNATE - Relational Persistence for Idiomatic Java**. Documento Técnico. 2007.

INSTITUTE OF MEDICINE. *The computer-based patient record: an essential technology for health care*. Revised edition, Division of Health Care Services, Institute of Medicine, National Academy of Science, Washington, D.C., USA, 1997.

ISO/TR 20.514:2005 Technical Report - Health informatics -- Electronic health record -- Definition, scope and context. Disponível em:
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39525

JOHNSON, R. et al *Spring – java/j2ee Application Framework*. Documento Técnico. 2004.

JANSEN, W.; AYERS, R. **Guidelines on PDA Forensics: Recommendations of the National Institute of Standards and Technology**. Gaithersburg, 2004.

KOTZ, D.; CHEN, G. *A survey of contextaware mobile computing research*. Documento Técnico. 2006.

KUHNEN, A. **Protótipo de uma aplicação LBS utilizando GPS conectado em celular para consultar dados georeferenciados**. 61 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2003.

LEÃO, B.F. **O prontuário eletrônico: como chegar lá ?**. Informática Brasileira em Análise. Ano II. n.16. Agosto 1998. Online. Disponível em:
http://www.cesar.org.br/analise/n_16/frameanalisen_16.html.

PETRY, K; LOPES P. M. A.; WANGENHEIM, A.V. **Padrões para a Interoperabilidade na Saúde**. [artigo completo]. In X Congresso Brasileiro de Informática em Saúde; 2006 Out 14-18; Florianópolis. Anais. Florianópolis-SC, 2006. Disponível em:
<http://www.sbis.org.br/cbis/arquivos/961.pdf>

MALUTA, L. C. G. **Pesquisa e Projeto de uma Aplicação em Computação Móvel para Monitoramento de Pacientes em UTI**. Florianópolis: UFSC, 2001.

MARIN, H. de F.; MASSAD, Eduardo; AZEVEDO NETO, R. S.; **O Prontuário Eletrônico do Paciente na Assistência, Informação e Conhecimento Médico**. Núcleo de Informática em Enfermagem Universidade Federal de São Paulo. São Paulo: HF Marin, 2003.

MARTHA, A. S. et al **Clinic Web: PEP e interação com dispositivos móveis**. In: X Congresso Brasileiro de Informática em Saúde, Florianópolis, 2006.

MCDONALD, C.J., BARNETT, G.O. Medical-Record Systems. In: Shortliffe, E.H., Perreault, L.E. (eds). *Medical Informatics: Computer Applications in Health Care*. New York: Addison- Wesley Publishing, 1990. p.181-218.

MEZAROBA, W. F.; MENEGON, M. P.; NICOLEIT, E. R. **Registro Eletrônico de Paciente em uma UTI: Comunicação, Interação com Dispositivos Móveis e Previsão de Expansibilidade**. In: XI Congresso Brasileiro de Informática em Saúde - CBIS 2008, Campos do Jordão. Anais do XI Congresso da Sociedade Brasileira de Informática em Saúde, 2008.

MOURÃO, E. J. **PROJETO DE APLICAÇÕES PARA DISPOSITIVOS MÓVEIS**. Monografia (Graduação) - Ciência da Computação, Universidade de Fortaleza, Fortaleza, 2004.

MUCHOW, J. W. **Core J2ME: tecnologia & MIDP**. São Paulo: Pearson Educação, 2004.

MURAKAMI, A. **MobMed - Inovando a Prática Médica Através de Sistemas Móveis de Informação**. São Paulo: USP, 2003.

NEIRA, R. A. Q. et al **Como incorporar conhecimento aos sistemas de registro eletrônico em saúde?** In: XI Congresso Brasileiro de Informática em Saúde, 2008, Campos do Jordão. XI Congresso Brasileiro de Informática em Saúde, 2008.

OLIVEIRA, E. C. M. **Conhecendo a plataforma J2EE - um breve overview**. Disponível em: <http://www.linhadecodigo.com.br/ArtigoImpressao.aspx?id=333>, Maio, 2004.

PALUDO, L. **Um Estudo sobre as tecnologias java de desenvolvimento de aplicações móveis**. Monografia (Graduação) - Ciência da Computação, Universidade Federal de Santa Catarina, Florianópolis, 2003.

PEKUS Cons. e Desenvolvimento Ltda. **Dispositivos móveis**. São Paulo, 2002. Disponível em: <http://www.pekus.com.br/palmtops.htm>. Acesso em: 12 maio de 2009.

PRÁ JUNIOR, J. V. Dal; **Protótipo de coleta de dados para um sistema de rastreabilidade de suínos: Estudo de caso da COOPERALFA**; Chapecó: UNOCHAPECO, 2006.

RUFINO, N. M. O. **Segurança em Redes Sem Fio**. 2. ed. São Paulo: Novatec, 2005. 224 p.

SALOMÃO, P.; SIGULEM, D. **Utilização do Computador de Mão Integrado à Telefonia Celular no Atendimento Médico: Desenvolvimento de Sistema e Avaliação**. In: XVIII Congresso Brasileiro de Engenharia Biomédica, São José dos Campos, SP, 2002.

SAVAGE, P. *A Book that Changed my Practice - Problem oriented medical records*. BMJ, 2001. Disponível em: <http://www.bmj.com/cgi/content/full/322/7281/275>
SBIS; CFM. Manual de Certificação para Sistemas de Registro em Saúde. 2008. Acesso em 03 maio 2009. Disponível em:
http://www.sbis.org.br/certificacao/Manual_Certificacao_SBIS_CFM_Fase2_v3.1_Consulta_Publica.pdf.

SCHAEFER, C. **Protótipo de aplicativo para transmissão de dados a partir de dispositivos móveis aplicado a uma empresa de transportes**. Monografia (Graduação) - Ciência da Computação, Universidade Federal do Pará, Blumenau, 2004.

SCHMITT JUNIOR, Arno José. **Protótipo de front end de controle de acesso usando J2Me**. 70 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2004.

SILVA, R.L. **Aplicativo para representante comercial em dispositivo móvel (PDA) usando a tecnologia J2ME e banco de dados**. Blumenau: FURB, 2005.

SITTIG, D.F. **Advantages of Computer-Based Medical Records**. The Informatics Review. Janeiro, 1999. Disponível em:
<http://www.informatics-review.com/thoughts/advantages.html>

SOARES, M. D. **WLAN / PWLAN pode substituir a terceira geração (UMTS) ou será mais uma tecnologia?** Disponível em:
http://www.wirelessbrasil.org/wirelessbr/colaboradores/marilson/wlan/wlan_pvlan.html
nov. 2003.

SOARES, L. F. G.; LEMOS, G.; COLCHER, S. **Redes de computadores: das LANs, MANs e WANs às Redes ATM**, 2a. Edição (revisada e ampliada), Ed. Campus 1995.

STRACHOSKI, A. F. **Especificação de um Prontuário Eletrônico com Base em Normas de Construção de Registros Eletrônicos de Saúde e Padronização da Informação Médica**. 2005. 92 f. Trabalho de Conclusão de Curso. (Graduação em Ciência da Computação) - Universidade do Extremo Sul Catarinense.

SUN, *Microsystems*. **Mobile Information Device Profile (MIDP); JSR 37, JSR 118 Overview**. Disponível em: <http://java.sun.com/products/midp/overview.html>. 2007.

SUN, *Microsystems*. **Saiba mais sobre a tecnologia Java**. Acessado em:
http://www.java.com/pt_BR/about. 2007.

TANENBAUM, Andrews S.; **Redes de Computadores**, Campus, 1997.

THOFEHRN, C.; LIMA, W. C. **Prontuário Eletrônico do Paciente – A Importância da Clareza da Informação**. Outubro, 2007.

VAN GINNEKEN, A.M., MOORMAN, P.W. The Patient Record. In: van Bommel, J.H., Musen, M.A.(eds.). *Handbook of Medical Informatics*. Houten, the Netherlands: Bohn Stafleu Van Loghum, 1997. p.99-115.

VASCONCELLOS LUZ; A. R. de A. **O Prontuário Eletrônico de Paciente e a Segurança da Informação: Uma Abordagem Arquivística**. Universidade do Rio de Janeiro – UNIRIO. Rio de Janeiro, 2004.

WAEAGEMANN, P. *The Five Levels of Electronic Health Records*. M.D.Computing, v.13, n.3, 1996.

WEED L. L. Medical records that guide and teach. N Engl J Med 1968;12(21):652-657.

W3C. **World Wide Web Consortium**. Disponível em: <www.w3c.org>. Acesso em: 11 nov. 2008.