

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**DESENVOLVIMENTO DE APLICATIVOS PARA DISPOSITIVOS MÓVEIS
COMPATÍVEIS COM A TECNOLOGIA WAP**

**DESENVOLVIMENTO DE APLICATIVOS PARA DISPOSITIVOS MÓVEIS
COMPATÍVEIS COM A TECNOLOGIA WAP**

Orientador: Idelbrando Ferreira de Lima

**NATAL/RN
2004**

IDELBRANDO FERREIRA DE LIMA
FACULDADE NATALENSE PARA O DESENVOLVIMENTO
DO RIO GRANDE DO NORTE
CURSO DE BACHARELADO EM SISTEMA DE INFORMAÇÃO

DESENVOLVIMENTO DE APLICATIVOS PARA DISPOSITIVOS MÓVEIS COMPATÍVEIS COM A TECNOLOGIA WAP

Monografia apresentada ao Curso de Bacharelado em Sistema de Informação da Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte como requisito parcial para a obtenção do título em Bacharel em Sistema de Informação.

Orientador:

Aluizio Ferreira da Rocha Neto

NATAL/RN

2004

Catálogo na Publicação – Biblioteca FARN
Setor de Processo Técnico

Lima, Idelbrando Ferreira de

Estudo sobre as limitações dos dispositivos móveis compatíveis com a tecnologia WAP no desenvolvimento de aplicativos/Idelbrando Ferreira de Lima. - Natal: FARN, 2004.
51p.

Orientador: Aluízio Ferreira Neto (M.sc)

Monografia (Graduação em Sistemas de Informação)-

Faculdade Natalense Para o Desenvolvimento do Rio Grande do Norte.

1. Informática – Monografia 2. Dispositivos Móveis – Limitações – Monografia I.Título

RN/FARN/BC

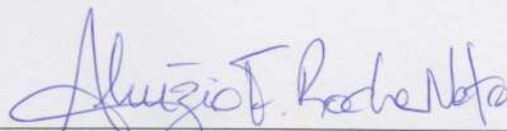
CDU 004

LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO
DO RIO GRANDE DO NORTE
CURSO DE BACHARELADO EM SISTEMA DE INFORMAÇÃO

**DESENVOLVIMENTO DE APLICATIVOS PARA DISPOSITIVOS MÓVEIS
COMPATÍVEIS COM A TECNOLOGIA WAP**

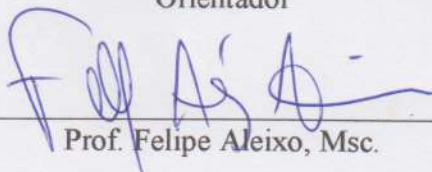
Idelbrando Ferreira de Lima

Monografia apresentada e aprovada em 13 de DEZEMBRO de 2004, pela
banca examinadora composta pelos seguintes membros.

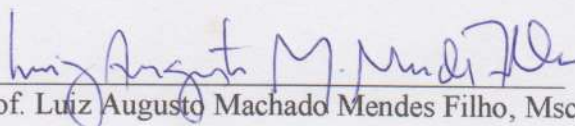


Prof. Aluizio Ferreira da Rocha Neto, Msc.

Orientador



Prof. Felipe Aleixo, Msc.



Prof. Luiz Augusto Machado Mendes Filho, Msc.

Natal, 13 de DEZEMBRO de 2004.

AGRADECIMENTOS

Agradeço a Deus por me dar forças e perseverança para não desistir diante das dificuldades no decorrer desses quatro anos, e por ter me concedido saúde. Agradeço a meus pais por terem me dado um bom exemplo de valores que são a educação e a oportunidade de adquirir novos conhecimentos em qualquer momento da vida e por me incentivando a seguir a minha carreira.

Agradeço a Sandra por ter sido minha amiga e minha família por sempre me apoiar e me encorajar. Agradeço ao professor Aluizio Rocha, meu orientador, por me ensinar a trabalhar e por me ensinar a lidar com a vida.

Catálogo na Publicação – Biblioteca FARN
Setor de Processo Técnico

Lima, Idelbrando Ferreira de

Estudo sobre as limitações dos dispositivos móveis compatíveis com a tecnologia WAP no desenvolvimento de aplicativos/Idelbrando Ferreira de Lima. - Natal: FARN, 2004.
51p.

Orientador: Aluizio Ferreira Neto (M.sc)

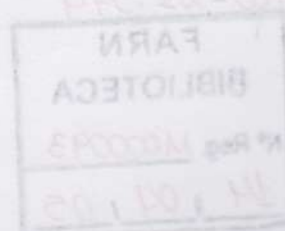
Monografia (Graduação em Sistemas de Informação)-

Faculdade Natalense Para o Desenvolvimento do Rio Grande do Norte.

1. Informática – Monografia 2. Dispositivos Móveis – Limitações – Monografia I.Título

RN/FARN/BC

CDU 004



Catálogo na Publicação - Biblioteca FARN
Setor de Processos Técnicos

Lima, Helibrando Ferreira de
- Estudo sobre as limitações dos dispositivos móveis compatíveis com a tecnologia WAP no desenvolvimento de aplicativos/relatório de Ferreira de Lima - Natal: FARN, 2004

Orientador: Aluizio Ferreira Neto (Msc)
Monografia (Graduação em Sistemas de Informação) -
Faculdade Natansense Para o Desenvolvimento do Rio Grande do Norte

I. Informática - Monografia 2. Dispositivos Móveis - I. Implicações - Monografia. I. Título

CDDU 004

RNFARN/RC

RD-21.374



M
004.738.5.057
L 696 d
2004

AGRADECIMENTOS

Agradeço a Deus por me dar forças e perseverança para não esmorecer diante das dificuldades no decorrer desses quatro anos, e por ter me concedido saúde. Agradeço a meus pais por terem me dado um bem muito valioso que foi a educação e a oportunidade de adquirir novos conhecimentos me apoiando durante os momentos difíceis e me incentivando a segui a diante.

A Sandra por ter sido sempre uma grande companheira durante todos os momentos, a meus amigos e minha família por entenderem que em alguns momentos não pude estar com eles. Agradeço a meus professores pela dedicação e empenho em passar os ensinamentos necessários para minha formação, e especialmente ao professor, Aluizio Rocha, meu orientador que muito me ajudou e procurou passar o conhecimento da melhor maneira possível.

A todos meus eternos agradecimentos.

RESUMO

O protocolo para aplicações sem fio (WAP) é o elemento chave que está levando o conteúdo da Internet para os telefones sem fio compatíveis com essa tecnologia. Com isso, surge a necessidade de se desenvolver aplicativos eficientes que possibilitem o acesso desses dispositivos aos serviços da Internet. Este trabalho realiza um estudo sobre as limitações dos dispositivos moveis compatíveis com (WAP) no desenvolvimento de aplicativos. O estudo apresenta os aspectos de usabilidade, como uma proposta de solução para resolver os problemas que as limitações dos dispositivos móveis acarretam.

Palavras Chaves: WAP, WML, WTP, PDA.

ABSTRACT

The protocol for wireless applications is the key element that is taking the market of smartphones for Internet. With that, arise the need to develop efficient applications that make possible access to mobile services in Internet. This work accomplishes a study on the general guidelines to develop applications that use the Wireless Application Protocol (WAP). The study presents the general guidelines of usability as a solution proposal to solve the problems that the limitations of the mobile devices cart. To access the data dynamically, were used common solutions in data access on Internet applications.

Key Word: WAP, WML, WTP, PDA.

2	REFERENCIAL TEÓRICO	23
2.1	ALINHAMENTO DE MARCAÇÃO WML	23
2.2	Elementos	23
2.3	Atributos	23
2.4	ENTENDIMENTO DAS LIMITAÇÕES	23
2.4.1	Tamanho máximo de tela	23
2.4.2	Resolução de Pixels limitadas	23
2.4.3	Limitação de Largura de Banda	23
2.4.4	Poder de Processamento e memória limitadas	23
2.4.5	Limitação de Formatos de Arquivos e Extensões de Arquivos	23
2.4.6	Problemas de Interoperabilidade de WML	23
2.5	ABORDAGEM DE ABORDAGEM	23
2.5.1	Uso de WML	23
2.5.2	Uso de aplicativos WAP	23
2.5.3	Problemas para desenvolvimento de aplicativos WML	23
2.6	UML	23
2.6.1	Análise de Requisitos	23
2.6.2	Caso de Uso	23
2.7	JAVA	23
2.8	JSR	23
2.9	ODBC	23
2.10	WAP OVER DATA	23
2.10.1	Entidade de Informação	23
2.10.2	Transferência de Dados	23
3	DESENVOLVIMENTO DE UM SISTEMA DE CONSULTAS PARA CELULAR WAP	23
3.1	REQUISITOS DO SISTEMA	23
3.2	REQUISITOS PARA O DESENVOLVIMENTO WML	23
3.3	MODELAGEM DO SISTEMA COM UML	23
3.3.1	Diagrama de Sequência - Diagrama de Caso de Uso	23
3.4	MODELAGEM DO BANCO DE DADOS	23
3.4.1	O Modelo Entidade-Relacionamento	23
3.4.2	Modelo Operacional de Dados	23
3.5	ANÁLISE DAS INTERAÇÕES DO SISTEMA	23
4	CONCLUSÃO	23
4.1	CONSIDERAÇÕES FINAIS	23
5	REFERÊNCIAS BIBLIOGRÁFICAS	23

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVOS	14
1.1.1	Objetivos Gerais	14
1.1.2	Específicos:	14
1.2	JUSTIFICATIVA	15
1.3	METODOLOGIA	16
1.3.1	Tipo de Pesquisa	16
1.4	ORGANIZAÇÃO DO TRABALHO	17
2	REFERENCIAL TEÓRICO	18
2.1	PROTOCOLO WAP	18
2.1.1	Componentes da arquitetura WAP	18
2.2	A LINGUAGEM DE MARCAÇÃO WML	21
2.2.1	Elementos	21
2.2.2	Atributos	21
2.3	OS DISPOSITIVOS WAP E SUAS LIMITAÇÕES	24
2.3.1	Tamanho reduzido da tela	25
2.3.2	Recursos de Entrada Limitados	25
2.3.3	Limitação da Largura de Banda	25
2.3.4	Poder de Processamento e memória limitados.	25
2.3.5	Limitações de Tamanho de Baralho e Elementos Gráficos.	26
2.3.6	Problemas de Interoperabilidade de WML	26
2.4	ASPECTOS DE USABILIDADE	27
2.4.1	Usuários Típicos	27
2.4.2	Usuários de aplicativos WAP	28
2.4.3	Objetivos para desenvolvimento de aplicativos usáveis	28
2.5	UML	31
2.5.1	Análise de Requisitos	32
2.5.2	Caso de Uso	32
2.6	JAVA	33
2.7	JSP	34
2.8	ODBC	35
2.9	BANCO DE DADOS	36
2.9.1	Entidade-Relacionamento	36
2.9.2	Linguagem SQL	37
3	DESENVOLVIMENTO DE UM SISTEMA DE CONSULTAS PARA CELULAR WAP	38
3.1	FUNCIONALIDADE	38
3.2	FERRAMENTAS PARA DESENVOLVIMENTO WML	38
3.3	MODELAGEM DO SISTEMA EM UML	38
3.3.1	Funcionalidade do Sistema - Diagrama de Casos de Uso	39
3.4	MODELAGEM DO BANCO DE DADOS	39
3.4.1	O Modelo Entidade-Relacionamento	39
3.4.2	Modelo Operacional de Tabelas	40
3.5	PROJETO DAS INTERFACES DO SISTEMA	41
4	CONCLUSÃO	46
4.1	CONSIDERAÇÕES FINAIS	47
	REFERÊNCIAS BIBLIOGRÁFICAS	48

LISTA DE FIGURAS

FIGURA 1: ARQUITETURA WAP	19
FIGURA 2: PROTOCOLOS EM CAMADAS DO WAP	19
FIGURA 3: CARACTERES PARA CONTROLE DE FORMATAÇÃO	22
FIGURA 4: CÓDIGO WML PARA REPRESENTAR CARACTERES COM ACENTUAÇÃO	23
FIGURA 5: CASOS DE USO PARA USUÁRIO	39
FIGURA 6: DIAGRAMA DE ENTIDADE E RELACIONAMENTOS	40
FIGURA 7: TABELAS DO SISTEMA E SEUS RELACIONAMENTO	40
FIGURA 8: TELA DE <i>LOGIN</i> DO USUÁRIO	41
FIGURA 9: TELA DE SELEÇÃO DAS DISCIPLINAS	43
FIGURA 10: TELA PARA SELEÇÃO DA UNIDADE	44
FIGURA 11: TELA DE CONSULTA DAS NOTAS E FALTAS	45

LISTA DE ABREVIATURAS E LISTA DE QUADROS

QUADRO 1: ARQUIVO WML "ALÔ MUNDO"	24
QUADRO 2: EXEMPLO DE PÁGINA JSP	35
QUADRO 3: FUNCIONALIDADE DO CASO DE USO DO USUÁRIO	39
QUADRO 4: PÁGINA WML DE LOGIN DO USUÁRIO	42
QUADRO 5: PARTE DO CÓDIGO DE CONSULTA AO BANCO	43
QUADRO 6: PARTE DO CÓDIGO WML DA PÁGINA PARA SELEÇÃO DAS UNIDADES	44
QUADRO 7: CÓDIGO WML COM RESULTADO DA CONSULTA DE NOTAS E FALTAS	45

TCP - Transport Control Protocol

TLS - Transport Layer Security

UDP - User Datagram Protocol

UML - Unified Modeling Language

URL - Uniform Resource Locator

XML - Extensible Markup Language

WAE - Wireless Application Environment

WAP - Wireless Application Protocol

WDP - Wireless Datagram Protocol

WML - Wireless Markup Language

WMLS - Wireless Markup Language Script

WSP - Wireless Session Protocol

WSPB - Wireless Session Protocol Binary

WTA - Wireless Telephony Application

WTAI - Wireless Telephony Application Interface

WTLS - Wireless Transport Layer Security

WTP - Wireless Transaction Protocol

WWW - World Wide Web

LISTA DE ABREVIATURAS E SIGLAS

CGI - Common Gateway Interface

DTD - Document Type Definition

HTML - Hyper Text Markup Language

HTTP - Hyper Text Transfer Protocol

JSP - Java Serve Pages

PDA - Personal Digital Assistant.

TCP - Transport Control Protocol

TLS - Transport Layer Security

UDP - User Datagram Protocol

UML - Unified Modeling Language

URL - Unit Resource Locator

XML - Extensible Markup Language

WAE - Wireless Application Environment

WAP - Wireless Application Protocol

WDP - Wireless Datagram Protocol

WML - Wireless Markup Language

WMLS - Wireless Markup Language Script

WSP - Wireless Session Protocol

WSPB - Wireless Session Protocol Browsing

WTA - Wireless Telephony Application

WTAI - Wireless Telephony Application Interface

WTLS - Wireless Transport Layer Security

WTP - Wireless Transaction Protocol

WWW - World Wide Web

1 INTRODUÇÃO

A Internet constitui um universo novo onde existe uma grande disponibilidade de informações e serviços. A tecnologia *Wireless* (sem fio) permite que dispositivos móveis como telefone celular, palmtop, PDAs (*Personal Digital Assistants* – Assistentes Digitais Pessoais), etc., possam se conectar com a Internet.

WAP sigla de *Wireless Application Protocol* (protocolo para aplicações sem fio) é um padrão que foi definido por um conjunto de grandes empresas de telefonia móvel para ser usado em aplicações baseadas na Internet, permitindo que aparelhos portáteis compatíveis com essa tecnologia acessem as informações e serviços da rede mundial de computadores.

Muitos projetos de telefones celulares são bastante complicados e contêm recursos que dificilmente são usados. Os diferentes aparelhos (WAP) existentes no mercado e suas limitações apresentam sérias implicações no modo como deve ser projetado um aplicativo para esses dispositivos que tenha um certo grau usabilidade, tornando-se necessário o conhecimento de suas limitações por parte dos usuários dessa tecnologia.

Faz-se necessário definir o termo usabilidade, o qual indica o grau de facilidade de utilização de um sistema por parte do usuário.

Embora os aspectos de usabilidade, independente de navegador sejam difícil de obter, o estudo aqui apresentado não tem o propósito de testar os diferentes navegadores existentes, tendo em vista sua grande variedade, mas sim, de apresentar uma pesquisa sobre as limitações dos dispositivos WAP.

Qualquer discussão prática detalhada dos problemas de usabilidade precisa ocorrer no contexto dos dispositivos específicos. Essa não é uma tarefa fácil, pois muitos dispositivos compatíveis com WAP estão chegando no mercado e no futuro próximo, mais ainda vão ser lançados.

1.1 OBJETIVOS

O WAP não é destinado a oferecer o conteúdo da Internet tal como o conhecemos, mas sim possibilitar que aplicações na rede mundial de computadores possam ser acessadas por usuários de aparelhos portáteis. O mesmo provê um protocolo com exibições muito simplificadas, próprias para telas pequenas com recursos de entrada limitados e baixa velocidade de comunicação.

A linguagem utilizada no WAP é WML (*Wireless Markup Language* – Linguagem de Marcação para Tecnologia Sem Fio). A WML especifica o formato necessário para o desenvolvimento de páginas WAP.

1.1.2 Especificar:

- Estudar e analisar as limitações das aplicações WAP.
- Apresentar as vantagens e benefícios do protocolo WAP, que pode ser utilizado como futuro padrão para operadores, fabricantes de dispositivos e desenvolvedores de aplicações, para transpor a lacuna entre a Internet e o domínio das redes sem fio.
- Identificar problemas relacionados com o desenvolvimento de aplicações.
- Aplicar os aspectos de usabilidade no desenvolvimento de aplicativos WAP.

1.1 OBJETIVOS

1.1.1 Objetivos Gerais

A pesquisa tem por objetivo apresentar um estudo sobre as limitações dos dispositivos WAP no desenvolvimento de aplicativos, e apontar os aspectos de usabilidade como proposta de solução dos problemas ocasionados por essas limitações.

Desenvolver uma aplicação para dispositivos móveis compatíveis com a tecnologia WAP, para avaliar os problemas ocasionados por suas limitações.

1.1.2 Específicos:

- Estudar e analisar as limitações das aplicações WAP.
- Apresentar as vantagens e benefícios do protocolo WAP, que pode ser utilizado como futuro padrão para operadoras, fabricantes de dispositivos e desenvolvedores de aplicação, para transpor a lacuna entre a Internet e o domínio das redes sem fio.
- Identificar problemas relacionados com o desenvolvimento de aplicações
- Aplicar os aspectos de usabilidade no desenvolvimento do aplicativo WAP.

1.2 JUSTIFICATIVA

O protocolo WAP está inserido, em meio a duas tecnologias que estão em desenvolvimento crescente: as comunicações móveis e a Internet. WAP é uma tecnologia usada para transpor a lacuna entre a Internet e o domínio das redes sem fio.

A comunicação de dados e acesso à Internet através de celulares e aparelhos portáteis, tem se tornado cada vez mais evidente em nosso dia-a-dia. O WAP pelas características de mobilidade no acesso às informações, é o elemento chave que está levando o grande mercado de telefones sem fio e aparelhos portáteis para a Internet. Esses pequenos dispositivos são bem equipados para realizar transmissão de voz de alta qualidade, facilitando o acesso a serviços da Internet através de micronavegadores WAP embutidos.

Os telefones sem fio estão mudando rapidamente a maneira com que as pessoas se comunicam no mundo. As aplicações WAP na Internet estão apenas na sua fase inicial. É uma tecnologia recente que certamente passará por amadurecimentos e inovações. Entretanto, o mais importante, do ponto de vista de um desenvolvedor, é que, mesmo na sua forma atual, a tecnologia WAP é uma poderosa ferramenta a ser explorada.

Os aplicativos WAP são a chave para fazer telefones inteligentes que sejam, realmente, inteligentes. Não é o hardware que realmente resolve os problemas do usuário ou fornece a estes recursos avançados e inteligentes. É o software que faz isso. Nesse sentido, cabe aos desenvolvedores de aplicativos WAP projetar aplicativos que sejam fáceis de usar e atendam às expectativas de seus usuários.

1.3 METODOLOGIA DO TRABALHO

Trata-se de uma pesquisa teórica e prática que a partir da análise dos conceitos de redes sem fio e das limitações dos dispositivos móveis, além do desenvolvimento de uma aplicação para tais dispositivos, proporciona uma visão ampla destas tecnologias e como as mesmas podem ser empregadas nas empresas como recursos de TI ou no nosso cotidiano nas tarefas usuais.

1.3.1 Tipo de Pesquisa

Pesquisa bibliográfica, explorativa, com a finalidade de estudar as limitações dos aparelhos da tecnologia estudada, assim como pesquisar e analisar os aspectos de usabilidade para solução dos problemas ocasionados por essas limitações. A pesquisa esta fundamentada em alguns autores da área, mostrando o que já foi estudado sobre o assunto e em que a pesquisa poderá auxiliar a comunidade acadêmica.

O método desta pesquisa é hipotético dedutivo funcional.

A literatura para estudo será adquirida no acervo da academia e também na Internet. Do ponto de vista prático, os dados serão coletados através dos resultados obtidos com o desenvolvimento do protótipo, todo o estudo será fundamentado em livros e documentação sobre o assunto pesquisado.

1.4 ORGANIZAÇÃO DO TRABALHO

O trabalho está organizado da seguinte forma: o primeiro capítulo possui a introdução da pesquisa, os objetivos esperados, metodologia, justificativa da realização dessa pesquisa e como esta organizado o texto.

O segundo capítulo, é um estudo sobre arquitetura WAP, as limitações dos dispositivos sem fio, e apresentação do referencial teórico das ferramentas para o desenvolvimento de aplicações com acesso dinâmico a site WAP.

No terceiro capítulo, é apresentado o projeto de um site WAP para acesso dos alunos da Farn para consultar suas notas e faltas.

No quarto capítulo, são feitas as análises dos resultados obtidos com a pesquisa e a implementação do projeto.

Por fim, o quinto capítulo apresenta as considerações finais deste trabalho monográfico.

2.1.1 Componentes da arquitetura WAP

O protocolo WAP é um de uma série de especificações para desenvolver aplicações Web para ambiente de redes móveis, é utilizado para acessar recursos em um de pequenos dispositivos.

Com as inovações tecnológicas, o WAP surge como uma ferramenta importante. Para programadores e pesquisadores.

Segundo Silveira (2000), a arquitetura WAP é formada por um ambiente executável e extensível para o desenvolvimento de aplicações para os dispositivos de comunicação móvel. Tem se baseia no modelo de protocolo WAP, que é composto por várias camadas de subprotocolos, e que cada camada pode ser acessada pela camada superior por outros serviços e aplicações.

A Figura 1 mostra a estrutura de funcionamento do protocolo WAP.

2 REFERENCIAL TEÓRICO

O referencial teórico aqui apresentado é fundamentado em alguns autores cuja obra fornece um embasamento científico necessário para o desenvolvimento da pesquisa.

2.1 PROTOCOLO WAP

O protocolo WAP foi iniciado em junho de 1997, com a união das companhias Ericsson, Nokia, Motorola e Phone.com, em uma associação de indústrias. Tem suas especificações abertas, o que significa que são discutidas e projetadas e ficam sob a administração de um consórcio WAP Fórum.

Segundo Oliveira (2000), WAP especifica a estrutura e os protocolos de rede para dispositivos móveis. As especificações são amplas e atinge todas as tecnologias de redes sem fio (*Wireless Network*) e as tecnologias Internet (tais como XML, URLs, Script, etc.). A meta é dar suporte tecnológico aos operadores, fabricante e desenvolvedores, permitindo-os a execução e implementação de serviços na rede móvel, de maneira rápida e flexível.

2.1.1 Componentes da arquitetura WAP

O protocolo WAP trata de uma série de especificações para desenvolver aplicações Web para ambiente de redes móveis, é otimizado para desafios elevados no uso de pequenos dispositivos.

Com as inovações tecnológicas, o WAP surge como uma ferramenta importante. Para programadores e pesquisadores.

Segundo Silveira (2000), a arquitetura WAP é formada por um ambiente escalável e extensivo para o desenvolvimento de aplicações para os dispositivos de comunicação móvel. Isto se realiza por meio do protocolo WAP, que é composto por várias camadas de subprotocolos, e que cada camada pode ser acessada pela camada superior por outros serviços e aplicações.

A Figura 1 mostra o esquema de funcionamento do protocolo WAP.

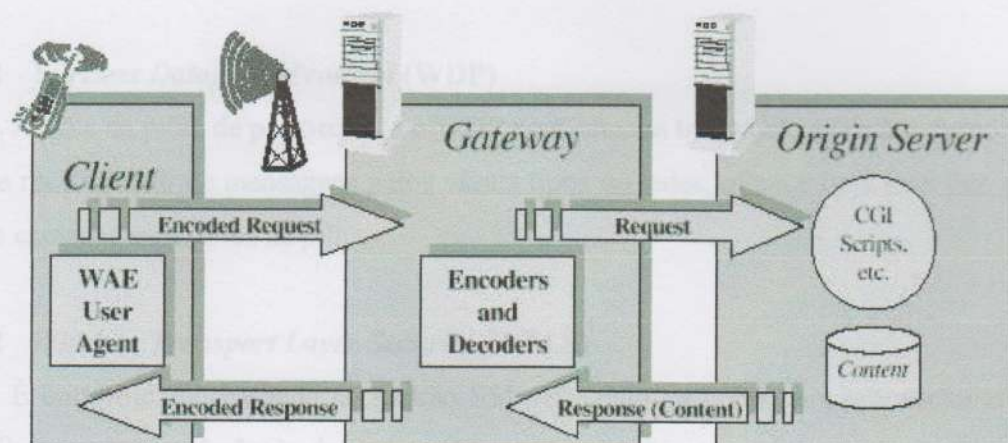


Figura 1: Arquitetura WAP

O WAP, na verdade, é formado por uma camada de protocolos, semelhante ao modelo de 7 camadas da ISO. Temos as camadas de transporte, segurança, transação, sessão e aplicação. A Figura 2 mostra a pilha de protocolos que formam o WAP.

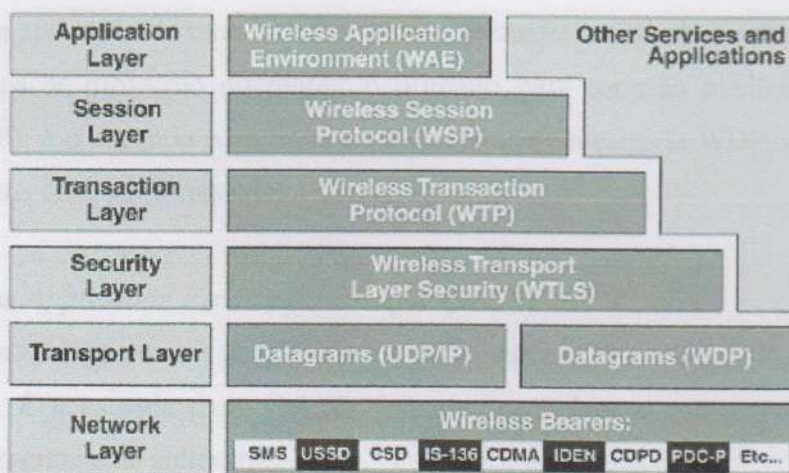


Figura 2: Protocolos em camadas do WAP

2.1.1.1 Wireless Datagram Protocol (WDP)

A base da pilha de protocolos é o WDP essa camada transporta os dados, responsável pelo envio e recebimento de mensagens pelos vários tipos de redes, oferece uma interface consistente com as camadas superiores da pilha.

2.1.1.2 Wireless Transport Layer Security (WTLS)

É um protocolo baseado no padrão SSL, utilizado na WEB para proporcionar segurança na realização de transferência de dados. Este protocolo foi especialmente desenvolvido para os protocolos de transporte WAP.

2.1.1.3 Wireless Transaction Protocol (WTP)

É um protocolo simplificado, idealizado para situações de baixa largura de banda, caso específico das comunicações sem fio.

2.1.1.4 Wireless Session Protocol (WSP)

Permite a aplicação de uma interface para o estabelecimento de sessões. Para isso foram criados dois tipos de protocolo diferentes: o primeiro para conexão mediante os serviços de transações (WTP), e o segundo permite acessar diretamente a camada WDP, sem a necessidade de estabelecer uma conexão direta com as outras camadas.

2.1.1.5 Wireless Application Environment Layer (WAE)

Aplicação de propósito geral baseada na combinação das tecnologias WWW e de telefonia móvel. É a camada mais alta ela fornece um ambiente que permite o uso de uma quantidade muito grande de aplicações.

De acordo com Demétrio (2000), a arquitetura de camadas WAP permite a outros serviços e aplicações utilizar as funções da pilha de camadas WAP por meio de interfaces bem definidas. Além disso as aplicações externas podem acessar diretamente as camadas sessão, transação, segurança e transporte.

2.2 A LINGUAGEM DE MARCAÇÃO WML

Segundo Oliveira (2000), WML (*Wireless Markup Language*) é uma linguagem que descreve o conteúdo WAP que será mostrado ao usuário. Com WML, podem-se mostrar informações no telefone celular, fornecer ao usuário opções de entrada de dados, e especificar como o dispositivo responderá quando o usuário ativar funções da interface ou pressionar o botão.

2.2.1 Elementos

A WML é definida por um conjunto de elementos. Cada elemento tem uma instrução única de código-fonte em letras minúsculas que pode ter uma de duas formas. A primeira forma é idêntica a HTML, e a segunda forma é usada quando os elementos não contêm conteúdo visível ou quando um elemento que pode ter conteúdo está vazio, todos elementos são delimitados pelos caracteres `<` e `>`: **<elemento> Minha Informação</elemento>**.

De acordo com Oliveira (2000) pode haver obrigatoriamente um elemento "documento" que é a raiz da hierarquia. Com o WML, o elemento "documento" é **<wml>**. Todos outros elementos que descrevem o WML estarão contidos nele. Os elementos são itens estruturados e são representados pelas tags. Os elementos podem também conter sub elementos.

2.2.2 Atributos

Cada elemento WML tem zero ou mais atributos que descrevem vários aspectos do elemento. Os atributos são declarados adicionando-se uma lista de palavras-chave em letras minúsculas e seus valores, entre apóstrofes ou aspas, à tag inicial. Os pares atributos/valores aparecem depois do rótulo da tag de abertura.

WML distribui o conteúdo de interface do usuário que é mostrado em uma ampla variedade de dispositivos. Na prática há muita variação da implementação para representar o código do aplicativo entre os diferentes tipos de navegadores. Infelizmente um clique de "distância" em um dispositivo em particular podem ser dois ou mais clique em outro dispositivo.

WML oferece suporte a texto e a imagem através de uma variedade de comandos de formatação e layout. O atributo **format** do elemento **<input>** permite definir uma string de controle para monitorar o que o usuário pode entrar em um campo de entrada de dados. A figura 3 apresenta alguns caracteres válidos para controle da formatação.

Formato	Caracteres de entrada permitidos
A	Caracteres alfabéticos maiúsculos ou pontuação
a	Caracteres alfabéticos minúsculos ou pontuação
N	Numéricos
X	Letras maiúsculas
X	Letras minúsculas

Figura 3: Caracteres para controle de formatação

As unidades básicas mostradas (uma por vez) no display do dispositivo sem-fio, são chamados de *CARDs*, esses são agrupados numa estrutura chamada *DECK*. Um *deck* é similar á uma página HTML identificado por uma URL e é a unidade de transmissão de conteúdo.

WML oferece suporte para gerenciamento da navegação entre *cards* e *decks*, e inclui comandos para manipulação de eventos que podem ser usados para navegação ou execução de scripts. WML também suporta *hyperlinks*.

Parâmetros podem ser configurados para todos os *decks* WML usando modelo de estados. Variáveis podem ser usadas em lugar de *strings* e são substituídas em tempo de execução. A configuração de parâmetros permite que recursos de redes sejam utilizados de maneira mais eficiente.

Para garantir a apresentação Correta do caractere, com acentuação é recomendada a utilização da codificação WML para os caracteres ASCII, a figura 4 apresenta os principais caracteres utilizados na língua portuguesa.

Código WML	Caractere ASCII
á	Á
â	Â
ã	Ã
ç	Ç
é	É
ê	Ê
í	Í
ó	Ó
ô	Ô
õ	Õ
ú	Ú
ü	Ü

Figura 4: Código WML para representar caracteres com acentuação

Todo o conteúdo WML é transmitido de forma compilada e codificada nas redes sem fio, essa compilação é feita pelo WAP Gateway, após o recebimento dos dados de uma requisição HTTP e da verificação da sintaxe do mesmo.

- Memória Ram e poder de processador limitado;
- Suporte a elementos gráficos limitado;
- Tamanho de arquivo limitado.

A seguir, no Quadro 1, temos um exemplo de código WML "Alo Mundo".

Quadro 1: Arquivo WML "Alô Mundo"

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
<card>
<p>
Alô Mundo !
</p>
</card>
</wml>
```

2.3 OS DISPOSITIVOS WAP E SUAS LIMITAÇÕES

Quando comparados aos computadores pessoais, os dispositivos WAP têm várias limitações, que trazem sérias implicações no modo como deve ser desenvolvido o aplicativo WAP. Aqui estão algumas delas:

- Tamanho reduzido da tela
- Recursos de entrada limitados;
- Limitação da largura de banda;
- Memória Ram e poder do processador limitado;
- Suporte a elementos gráfico limitado;
- Tamanho de baralho limitado.

2.3.1 Poder de Processamento e memória limitados.

Apesar de alguns dispositivos WAP terem mais recursos que outros em termos de memória e processamento, o fato é que grande parte destes são celulares móveis com poder muito limitado. Essas observações devem ser analisadas durante o projeto de aplicativos WAP.

2.3.1 Tamanho reduzido da tela

As telas dos dispositivos são pequenas. O desenvolvedor deve adotar como regra geral o planejamento de uma interface com o usuário de tamanho reduzido pensando em termos de 4 linhas de 15 a 18 caracteres. De acordo com Arehart et al (2000), se o desenvolvedor tomar essas precauções, seus aplicativos poderão ser acessados sem problemas pela maioria de seus usuários, como também serão usáveis em dispositivos com telas maiores. Por outro lado, os aplicativos desenvolvidos para tela grande tendem a aparecer mal em uma tela pequena.

2.3.2 Recursos de Entrada Limitados

Atualmente a maioria dos dispositivos móveis é constituída de telefones, os quais possuem um teclado específico para esse aparelho. A introdução de texto torna-se uma tarefa difícil. Isso aumenta a probabilidade de os usuários navegarem com o uso apenas de portais, incluído portais personalizados, em que você pode especificar minuciosamente o tipo de conteúdo que deseja acessar.

2.3.3 Limitação da Largura de Banda

Os dispositivos WAP tem pouca largura de banda disponível se comparados com os PCs na Internet. Que podem utilizar uma largura de banda mais ampla. Segundo Arehart et al (2000), Mesmo no futuro, quando as telecomunicações viram a portadoras de largura de banda alta, as pessoas podem fazer uso de soluções menos dispendiosas como uma largura de banda baixa. Portanto os desenvolvedores WAP, precisam ter essa visão, ou seja, mesmo com maior largura de banda, os dados podem vir em pequenos trechos para os usuários. É necessário desenvolver aplicativos que estejam em sintonia com esse cenário.

2.3.4 Poder de Processamento e memória limitados.

Apesar de alguns dispositivos WAP serem mais poderosos que outros em termos de memória e processamento, o fato é que grande parte destes será constituída de telefones móveis com poder muito limitado. Essas observações devem ser analisadas durante o projeto de aplicativos WAP.

2.3.5 Limitações de Tamanho de Baralho e Elementos Gráficos.

Um obstáculo importante para o desenvolvimento de aplicativos WAP usáveis e a interoperabilidade entre os diferentes dispositivos compatíveis com WAP que existem no mercado e como a linguagem WML será implementada para se obter uma boa interface, observando que existem vários tipos de navegadores.

2.3.6 Problemas de Interoperabilidade de WML

Independentemente do bom suporte ou não nas especificações WAP, o conteúdo e as interfaces dos aplicativos precisam ser bem apresentados para o usuário. De acordo com Arehart et al (2000), na prática, a implementação para representar o código WML entre os diferentes tipos de navegadores sofre uma variação nos vários navegadores existentes. Infelizmente para acessar uma opção em um dispositivo em particular segue-se uma seqüência de passos que pode variar em outro dispositivo diferente acessando a mesma opção no mesmo aplicativo.

Essa situação não é simples de resolver de acordo com Arehart et al (2000): "Implementar várias versões de um aplicativo (uma para cada família de navegadores) pode ser a melhor opção disponível. Entretanto existem outras maneiras de contornar esse problema".

Uma observação importante é a diferença entre WAP e WEB em relação ao público que ambas atraem. Enquanto a WEB atrai um público de diversas partes do mundo, o WAP freqüentemente é utilizada por assinantes de uma rede de telecomunicações específica, em um país específico. Tendo uma idéia de como será o suposto público-alvo, o desenvolvedor pode projetar o sistema observando as características dos micronavegadores e *gateways* que por ventura poderão ser usados.

- * Evento de dispersão no ambiente que os cerca.
- * Suas experiências e habilidades com aplicativos similares.
- * O que esperam do aplicativo.
- * Quais são suas vontades.
- * Em qual contexto geográfico estão inseridos.

2.4 ASPECTOS DE USABILIDADE

A qualidade de um software é influenciada por vários fatores, entre os quais, destacamos a usabilidade. De acordo com Arehart et al (2000), para avaliar o quanto um aplicativo WAP é usável deve ser observado alguns aspectos como:

- As habilidades intelectuais exigidas para aprender a usar o sistema.
- O tempo exigido para aprender a usar o sistema
- O quanto o sistema é intuitivo.
- O nível de frustração envolvido no uso do sistema.

No desenvolvimento de aplicações WAP é fundamental a prática das regras de usabilidade, e uma observação importante é:

Avaliar o valor da interface com o usuário de um sistema tem se tornado cada vez mais importante com o crescimento do uso de computadores, de tal modo que agora existe um campo da informática, a interação homem-máquina (HCI, Human-Computer Interaction), que trata especificamente da construção de sistemas usáveis. (AREHART, et al. 2000 p.183)

2.4.1 Usuários Típicos

Arehart et al (2000), afirma que “quando você esta construindo um aplicativo de qualquer tipo não apenas WAP deve-se fazer a seguinte pergunta:” Quem é o usuário típico de meu aplicativo?. Isso é muito importante, pois o primeiro passo da construção de aplicativos fáceis e intuitivos é entrar na mente do usuário típico.

Usuários típicos são aqueles que têm seu perfil traçado levando em consideração alguns aspectos como:

- Evento de dispersão no ambiente que os cerca.
- Suas experiências e habilidades com aplicativos similares.
- O que esperam do aplicativo.
- Quais são suas motivações.
- Em qual contexto geográfico estão inseridos.

2.4.2 Usuários de aplicativos WAP

Entender quem são seus usuários é um passo importante no projeto de um aplicativo. Fazer um perfil de seu usuário e com base nesse perfil procurar entender suas motivações ajudará o desenvolvedor a estruturar seu aplicativo e a decidir quais recursos devem ser incluídos ou descartados. Embora esses passos sejam importantes para qualquer aplicativo, trata-se de uma necessidade absoluta no contexto WAP. Arehart et al (2000) enfatiza que “os usuários de WAP não ficam diante de um computador pessoal. Eles estão em movimento... Os usuários de WAP estão sujeitos a muitos eventos de dispersão no ambiente que os cerca”.

Todas as limitações dos dispositivos móveis devem ser lembradas constantemente, durante todas as fases do processo de desenvolvimento de aplicativos WAP. Segundo Arehart et al (2000), a fase de projeto é mais importante do que a fase de desenvolvimento em aplicativos WAP, pois um mau projeto com certeza compromete o sucesso da operação inteira.

2.4.3 Objetivos para desenvolvimento de aplicativos usáveis

Segundo Arehart et al (2000) “queremos atingir nossos objetivos rápida e eficientemente possível, mas, infelizmente, a facilidade de uso pelo usuário e uma eficiência ótima são - se não mutuamente exclusivas - um compromisso”. Podem-se destacar alguns objetivos quanto ao desenvolvimento de aplicativos que os usuários achem agradável de usar, são eles:

2.4.3.1 Aplicativos Intuitivos

Interfaces intuitivas são mais fáceis para os usuários utilizarem e, mais provavelmente, atrairão o usuário de volta para uma experiência interativa mais agradável. Os usuários precisam usar seu aplicativo sem que haja a necessidade de consultar o manual da aplicação para realizar operações no software consideradas básicas.

Arehart et al (2000) enfatizam que “os usuários tendem a usar apenas uma pequena fração da funcionalidade de um aplicativo. Às vezes, contudo, eles tentam executar operações ligeiramente menos comuns e devem conseguir fazer isso sem a ajuda externa de manuais e outros usuários”.

2.4.3.2 Aplicativos Eficientes

É importante desenvolver um aplicativo que seja o mais rápido de usar possível. Embora isso possa ser difícil em muitas situações, deve ser feito um balanço entre a qualidade de entrada do usuário e os resultados do aplicativo, ou seja, o aplicativo deve ser o mais eficiente possível. A maioria das pessoas é inerentemente impaciente até certo ponto.

2.4.3.3 Capacidade de Memorização

Embora um usuário possa ter de procurar um pouco para descobrir como executar uma operação incomum, é importante que, quando ele tiver descoberto, não tenha de passar pelo mesmo esforço novamente, quando quiser executar a mesma operação duas semanas depois. Ele deve se lembrar do processo facilmente. Se uma ação só é executada raramente, é melhor que seja omitida por completo.

2.4.3.4 Tolerância

Os usuários nunca devem se encontrar a um clique de uma operação destrutiva. Os aplicativos WAP devem ser tolerantes. É importante pedir confirmação para cada ação destrutiva para que, por uma falta de atenção, o usuário não perca o tempo gasto anteriormente até chegar ao estágio atual.

Após analisar alguns dos objetivos para desenvolvimento de aplicativos usáveis apresenta-se um conjunto de aspectos de usabilidade para atingi-los, são eles:

2.4.3.5 Classificação das atividades dos usuários

Consiste em identificar as principais atividades que a maioria dos usuários realizará e construir seu aplicativo de modo que permita que os usuários executem essas atividades da maneira mais rápida possível. De acordo com Arehart et al (2000) “você pode perder até metade de seus usuários para cada clique que eles tenham de dar. Além disso, se você exige entrada de texto, os estudos tem mostrado que 7 entre 8 usuários simplesmente param de usar o aplicativo”.

As atividades de usuário devem ser pensadas com muito cuidado. Para definir as principais funcionalidades dos aplicativos WAP, podemos começar tentando classificar as atividades do usuário em uma das três categorias a seguir:

- Atividades que a maioria dos usuários executam a maior parte do tempo em que acessam o aplicativo.
- As atividades que a maioria dos usuários executam ocasionalmente.
- As atividades que os usuários especializados executam de vez em quando.

De acordo com Arehart et al (2000) identificar e classificar as atividades do usuário é a base para dividir o fluxo de navegação do aplicativo e otimizar o caminho de navegação exigido para que sejam executadas as principais atividades da aplicação.

2.4.3.6 Minimização da Entrada de Dados

O teclado da maioria dos dispositivos WAP são poucos intuitivos e de difícil entrada de textos. O aplicativo deve exigir entrada de dados textuais somente quando for absolutamente necessário. Além disso, o modo de entrada do dispositivo deve ser configurado para suportar o formato esperado para os dados inseridos pelos seus usuários. Minimizar a entrada de dados textuais não é simples, vamos examinar algumas técnicas empregadas para conseguir isso.

- Uso de mecanismo de personalização para reconhecer usuários.
- Uso de listas de opções se puder, permitindo que o usuário faça escolhas e interaja com o aplicativo.
- Utilize mascaras de entrada, elas podem ser úteis em alguns casos, mas atenção: elas não são suficientemente poderosas para muitos tipos de validação.
- Informe ao usuário por meio de texto simples o formato da mascara de entrada.
- Uso de links para navegação.

2.4.3.7 Texto conciso

Como os dispositivos WAP têm telas pequenas, o texto deve ser curto, claro e informativo. Alguns dispositivos cortam textos longos em algumas situações, em vez de fazê-lo mudar de linha automaticamente, o que pode trazer consequências terríveis para a usabilidade, portanto é preciso evitar textos longos. Use linguagens dos usuários para os quais o aplicativo se destina, seja coerente no uso de termos.

2.5.1 Análise de Requisitos

2.4.3.8 Implementação de Funcionalidade Retrocesso.

Implemente funcionalidades que permitam que os usuários naveguem para trás, quando mudar de idéia, pois se o usuário segue um link e perde o contexto atual, a impossibilidade de voltar é percebida como uma armadilha. De acordo com Arehart et al (2000) “os usuários se sentem menos apreensivos usando um aplicativo que ofereça suporte a uma navegação para trás sólida”.

2.4.3.9 Sempre Testar os Aplicativos

O máximo possível de usuários devem utilizar o aplicativo. Isso informará o quanto a Interface do aplicativo é intuitiva. Se eles tiveram muitos problemas, então, isso implica que a interface não foi bem projetada sendo necessárias algumas modificações no aplicativo. De acordo com Arehart et al (2000) não devemos pular a fase de testes. Ela é muito importante, lembrando que as pessoas que testam seus aplicativos sempre devem representar usuários típicos.

2.5 UML

Para o desenvolvimento de aplicações para dispositivos móveis é fundamental o uso da UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada).

Modelamos para entender procedimentos complexos. O software, hoje, está mais complexo do que nunca. Como resultado, nós o modelamos. Nossos modelos descrevem o que queremos construir, o que estamos construindo e o que construímos. (Conallen, 2003, p.4)

De acordo com Conallen (2003), para a criação de aplicações eficiente, é fundamental modelar toda a lógica de negócios do sistema, independentemente de onde ou como ela esteja implementada no sistema.

2.5.1 Análise de Requisitos

Requisito é uma descrição das necessidades ou desejos para um produto. Nesta fase acontece a identificação e documentação do que tem realmente relevância para a modelagem, evitando a ambigüidade que poderia deturpar o entendimento do sistema por parte dos clientes e membros da equipe de desenvolvimento.

Durante a análise de requisitos, devem ser identificados os atores, que são os elementos externos que deveram interagir com o sistema, bem como as funcionalidades que o sistema deve oferecer.

2.5.2 Caso de Uso

Casos de Uso (do Inglês *Use Case*) são utilizados na construção do sistema para identificar por meio de cenários as regras do negocio. Um cenário é definido por Conallen (2003) como "uma seqüência de passos que descreve uma interação entre um usuário e um sistema".

Este cenário é uma alternativa que pode acontecer. No entanto uma determinada ação pode não ser completada, então o passo seguinte já consiste em um outro cenário diferente.

Conallen (2003) enfatiza que casos de uso "são uma excelente forma de entender o ponto de vista do usuário simplesmente pelo fato de modela o que ele precisa executar". Tendo em vista que e no caso de uso que são apresentadas de maneira simplificada as várias seqüências de ações de um ator (Agente Externo) que utiliza o sistema para completar o processo.

Segundo Conallen (2003) Para que um sistema seja visto como uma aplicação WEB é preciso haver uma regra de negócio no servidor. Se não houver nenhuma regra de negócio no servidor, então, o sistema não deverá ser visto como uma aplicação WEB. Com a inclusão de um servido de aplicação o sistema deve gerenciar o estado e a regra de negócio. A lógica de negócio é composta de regras e processos que formam o estado do negócio do sistema.

Para o autor, especificamente, a UML expressa modelos e projetos de sistema em um modo orientado a objetos, muito embora ela esteja sendo usada para descrever tipos de sistemas variando de processos e estruturas organizacionais genéricos de negócios a projetos incorporados de sistemas em tempo real.

Tendo um único modelo central de toda a lógica de negócio em um sistema, consegue-se compreendê-lo melhor e, eventualmente aperfeiçoá-lo em futuras versões do sistema. Conallen (2003, p.5) afirma que: “Os modelos freqüentemente conectam diferentes tipos de artefatos - especificações de caso de uso a tabela de banco de dados - e proporcionam uma cadeia de responsabilidade e rastreamento”. UML desempenha uma função muito importante na busca por falhas em sistemas de rede sem fio, com a modelagem podemos fazer o mapeamento de todo o projeto.

Conallen (2003) acentua que uma definição de tela de perfil do usuário pode ser rastreada até chegar ao código JSP que a implementa e o estado muda o comportamento de uma ordem de compra que pode ser rastreada até uma especificação de caso de uso que a define – tudo feito através dos vários modelos UML do sistema.

2.6 JAVA

A linguagem de programação Java foi criada pela Sun Microsystems. No início foi destinada a pequenos dispositivos eletrônicos. Suas características multiplataforma possibilitam a inclusão de pequenas aplicações diretamente nas páginas WEB, isso tudo acontece independente do sistema usado no dispositivo do cliente.

Os desenvolvedores de aplicativos WAP, têm na linguagem de programação Java um forte apoio para desenvolver suas aplicações. De acordo com Thompson (2002, p.18) “a linguagem Java foi projetada levando em conta o que havia de melhor e pior em outras linguagens de programação, se tornando uma linguagem muito útil nestes tempos de interconectividade e migração tecnológica constante”.

Java é uma linguagem de programação totalmente orientada a objetos. Seu funcionamento, no que diz respeito ao desenvolvimento, é semelhante a outras linguagens, é uma das poucas, senão a única linguagem de programação que permite ao programador criar um único código que roda sem alteração em qualquer computador que possua uma *Java Virtual Machine* instalada.

Quadro 2: Exemplo de página JSP

Um dos objetivos do java é permitir a elaboração de softwares que possa executar de forma independente em máquinas pequenas. O tamanho do interpretador básico e do suporte as classes é de cerca de 40KB; a adição de bibliotecas padrão básica e do suporte a subprocessos (essencialmente, um microkernel contido em si mesmo) acrescenta outros 175KB (HORSTMANN, Cay S; CORNELL, Gary 2001p.5).

2.7 JSP

Java Serve Pages (JSP) é uma tecnologia que serve a duas comunidades diferentes de programadores. Os designs de páginas usam a tecnologia JSP para adicionar poderosas capacidades de conteúdo dinâmico a sites da WEB e aplicações on-line. Os programadores de Java escrevem o código que implementa essas capacidades por detrás dos panos.

A JSP oferece várias vantagens como um sistema para geração de conteúdo dinâmico. Segundo Fields et al (2000), dentre estas vantagens, encontram-se a performance aperfeiçoada na CGI e um modelo de programação que enfatiza o design de aplicações centradas em componentes.

Esse modelo de programação permite que os programadores usem a JSP para manter uma rígida separação entre os elementos de apresentação básica. Esta separação, por sua vez, facilita uma divisão do trabalho dentro da equipe de desenvolvimento da WEB, ao estabelecer uma interface bem definida entre o desenvolvimento da aplicação e o design da página.

JSP interage com o servidor da WEB e o navegador do usuário final. As páginas JSP na verdade funcionam para produzir conteúdo dinâmico. As marcas JSP permitem que designs de páginas interajam com componentes Java através de marcas do tipo HTML ou WML, ao invés de código Java.

Segundo Arehart et al (2000) JSP é uma página WML que pode conter elementos JSP. Mesmo sem elementos JSP, ela ainda é uma página JSP legítima (basta mudar a extensão de WML para JSP) e será apresentada normalmente. No Quadro 2 é apresentado um exemplo de página em JSP.

Quadro 2: Exemplo de página JSP

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE>Contador JSP</TITLE>
</HEAD>
<BODY>
<!-- Declaração -->
<%! int n = 0; %>
<!-- Scriptlet -->
<% n = n + 1; %>
<!-- Expressão -->
Essa página foi acessada <%= n %> vezes.
</BODY>
</HTML>
```

2.8 ODBC

O padrão *Open Database Connectivity* (ODBC) Segundo Thompson (2002), é um padrão aberto de conectividade a banco de dados é uma interface com grande disponibilidade de drives para acesso a banco de dados. Existem drives para bancos de dados antigos, e mais atuais, como o MS Access, que também é instalado junto com o MS Access. Já o padrão Java de conectividade a banco de dados JDBC (*Java Database Connectivity*), é a interface que possibilita as aplicações Java acessarem banco de dados relacionais e demais arquivos de dados. É atualmente a forma mais pratica e rápida para conectar uma aplicação Java a um arquivo de dados.

Thompson (2002) Enfatiza que, embora um programa em Java possa utilizar diretamente a interface ODBC para acessar arquivos de dados, este procedimento é trabalhoso e elimina a portabilidade da aplicação. Não devemos esquecer que os *drivers* ODBC geralmente são escritos em linguagem C. Para superar este obstáculo, os programadores utilizam uma “ponte”, a JDBC-ODBC.

2.9 BANCO DE DADOS

Segundo Silberschatz et al (1999), um *Banco de Dados* é um conjunto de dados relacionados a um assunto ou propósito específico. Os registros de empregados em um arquivo, uma coleção de selos em um álbum, uma agenda com os nomes dos gerentes de vendas, cada um destes conjuntos é um banco de dados.

Um *Sistema Gerenciador de Banco de Dados* (SGBD) é um sistema que armazena e recupera informações em um banco de dados, ou seja, o arquivo de empregados, o álbum ou a agenda. Um SGBD computadorizado é um programa que pode ser utilizado para armazenar e recuperar os dados em seu computador.

O Microsoft Access é um *Sistema Gerenciador de Banco de Dados Relacional* (SGBDR), que armazena e recupera informações de acordo com os seus relacionamentos que você define. Com o Microsoft Access, você pode organizar os seus dados de acordo com o assunto, para que os dados sejam fáceis de verificar e controlar, e também pode armazenar informações de como os assuntos diferentes se relacionam para que seja fácil agrupar dados relacionados.

2.9.1 Entidade-Relacionamento

Silberschatz et al (1999), enfatiza que o modelo entidade-relacionamento é baseado na percepção do mundo real, que consiste em um conjunto de objetos básicos chamados entidades e nos relacionamentos entre esses objetos. O objetivo é facilitar o projeto de banco de dados, possibilitando a especificação da estrutura lógica geral do banco de dados.

2.9.2 Linguagem SQL

Segundo Silberschatz et al (1999), a *Structured Query Language* (SQL) é uma linguagem muito usada na consulta, atualização e gerenciamento de banco de dados relacionais.

Cada consulta criada no MS-Access tem uma instrução SQL associada que define as ações daquela consulta. Normalmente é mais conveniente usar a grade QBE e as ferramentas gráficas de estruturação de consulta do Access para criar e modificar consultas.

3.1 FUNCIONALIDADE

Sugere-se que o aluno da FARN deva verificar suas notas e faltas em qualquer hora durante seu curso. O aluno digita no celular WAP o endereço do site do sistema e digita após sua matrícula. O sistema valida e confirma e na página seguinte são mostrados os dados sobre os nomes das disciplinas do semestre vigente. Após a seleção do link desejado, são disponibilizadas para consulta, as notas e faltas na tela do micro navegador do celular.

3.2 FERRAMENTAS PARA DESENVOLVIMENTO WML

Os simuladores são ferramentas que têm a capacidade de simular a rede de celular em um computador. No desenvolvimento de aplicação foi utilizado o software UP.SDK 4.0 da Philips para criar o aplicativo e o IDE de nome UP.SDK para a criação do código. O UP.SDK é um software utilizado para realizar testes reais de funcionamento de um aplicativo. Esse software tem uma janela DOS que auxilia na criação e execução de programas durante o desenvolvimento de aplicações para dispositivos WAP.

3.3 MODELAGEM DO SISTEMA EM UML

Os atores que vão interagir diretamente com o sistema de consulta a notas e faltas são:

- **Usuário** - Qualquer acadêmico ou professor que deseja consultar suas notas e faltas, por meio de um aparelho celular WAP.

As principais funcionalidades do Sistema são:

- Visitar o site (Usuário).
- Login no site (Usuário).
- Consultar notas (Usuário).

3 DESENVOLVIMENTO DE UM SISTEMA DE CONSULTAS PARA CELULAR WAP

3.3.1 Funcionalidade do Sistema - Diagrama de Caso de Uso

Este capítulo aborda o desenvolvimento de um sistema de informação cuja interface é voltada para celulares WAP. O sistema é baseado em páginas JSP que acessam um banco de dados Access e exibem as informações em documentos WML.

3.1 FUNCIONALIDADE

Suponhamos que o aluno da FARN deseja verificar suas notas e faltas nas disciplinas do semestre em curso. O aluno digita no celular WAP o endereço do site, ao acessar a página digita sua matrícula. O sistema valida a matrícula e na página seguinte são mostrados os *links* com os nomes das disciplinas do semestre vigente. Após a escolha do *link* desejado, são disponibilizadas para consulta, as notas e faltas na tela do micro navegador do celular.

3.2 FERRAMENTAS PARA DESENVOLVIMENTO WML

Os emuladores são ferramentas que têm a capacidade de simular a tela do celular em seu computador. No desenvolvimento da aplicação foi utilizado o software UP.SDK 4.0 da phone.com para testar o aplicativo e o bloco de notas para a digitação do código. O UP.SDK é um software utilizado para realizar testes reais do funcionamento de um aplicativo. Esse software tem uma janela DOS que auxilia na análise e correção de problemas durante o desenvolvimento de aplicações para dispositivos WAP.

3.3 MODELAGEM DO SISTEMA EM UML

Os atores que irão interagir diretamente com o sistema de consulta a notas e faltas são:

- **Usuário** - Qualquer acadêmico ou professor que deseje consultar suas notas e faltas, por meio de um aparelho celular WAP.

As principais funcionalidades do Sistema são:

- Visitar o site (Usuário);
- Logar no site (Usuário);
- Consultar notas (Usuário);

A figura 5 apresenta a modelagem do caso de uso. O Quadro 3 informa a descrição do caso de uso.

3.3.1 Funcionalidade do Sistema - Diagrama de Casos de Uso

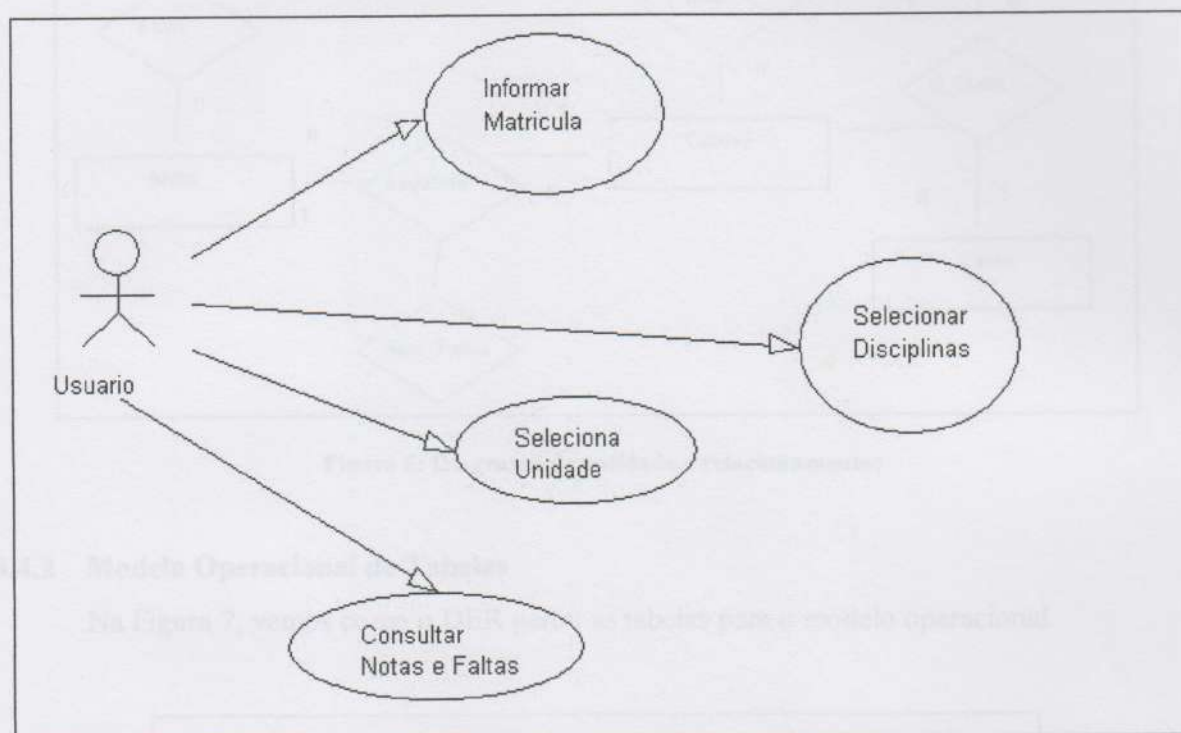


Figura 5: Casos de uso para Usuário

Quadro 3: Funcionalidade do caso de uso do Usuário

Use Case	Descrição
Login Usuário	Faz Login do usuário no sistema
Escolher Disciplina	Seleção da disciplina para consulta
Selecionar Unidade	Usuário seleciona primeira ou segunda unidade
Consultar Notas e faltas	Mostrar notas e faltas

3.4 MODELAGEM DO BANCO DE DADOS

3.4.1 O Modelo Entidade-Relacionamento

Na Figura 6, podemos ver o DER (Diagrama Entidade-Relacionamento) do modelo construído para a aplicação proposta.

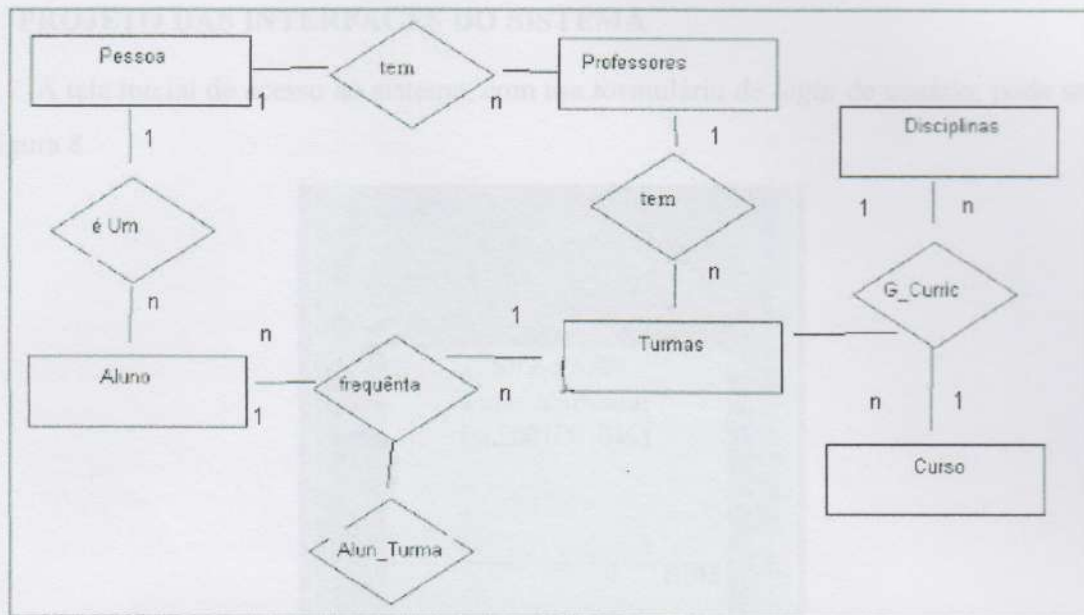


Figura 6: Diagrama de entidade e relacionamentos

3.4.2 Modelo Operacional de Tabelas

Na Figura 7, vemos como o DER gerou as tabelas para o modelo operacional.

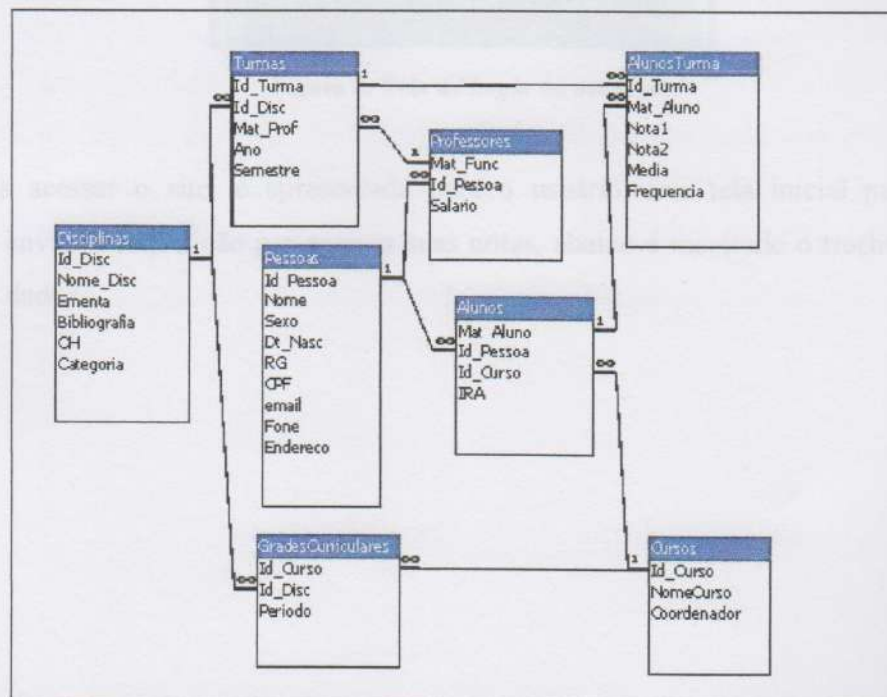


Figura 7: Tabelas do sistema e seus relacionamento

3.5 PROJETO DAS INTERFACES DO SISTEMA

A tela inicial de acesso ao sistema, com um formulário de *login* de usuário, pode ser vista na Figura 8.



Figura 8: Tela de *Login* do usuário

Após acessar o site, é apresentada para o usuário uma tela inicial para digitar sua matrícula e enviar a requisição pra acessar suas notas, abaixo é mostrado o trecho de código de entrada dos dados.

Quadro 4: Página WML de Login do usuário

```

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
  <card id="card1" >
    <do type = "accept" label = "Enviar">
      <go href="AlunosWap.jsp" method="post" >
        <postfield name="MATRICULA" value = "$MATRICULA" />
      </go>
    </do><p align="center"><i>SITE FARN</i><br/>
      <p>EnterMaricula:<br/>Ex.2001B010461
      <input name="MATRICULA" title="Matricula" emptyok="false"
        format="NNNNANNNNNN" maxlength="11" />
    </p>
  </card>
</wml>

```

Comentário, nas três primeiras linhas, o código especifica que as *tags* usadas no programa estão em conformidade com as especificações WAP. No código da pagina inicial para digitar a matricula, podem-se observar as *tags* <go> que contem o endereço da pagina AlunosWap.jsp e o método Post para o envio da matricula. Como objetivo para que o aplicativo seja fácil de usar foi utilizada uma interface simples e intuitiva para facilitar a entrada dos dados. Foi aplicado como aspecto de usabilidade a utilização de mascara de entrada pelo atributo *format="NNNNANNNNNN"* dentro da *tag* <input>, para formatação da matricula digitada, e mostrado na tela do dispositivo o formato dos dados exigidos para digitação.

A tela com *links* para as disciplinas que o aluno cursou é vista na Figura 9. A idéia desta tela é apresentar apenas as disciplinas que o aluno já cursou e que contém as notas que o aluno deseja obter.



Figura 9: Tela de seleção das disciplinas

Após a validação da matrícula é retornado a consulta, onde são mostradas as disciplinas para que o usuário selecione o link desejado.

Quadro 5: Parte do código de consulta ao banco

```
...
<select name="disc" value="0" title="Disciplinas:">
<% while(ResultSet.next()){ %>
<option value="<%= ResultSet.getString("Id_Disc")%>">
<%=ResultSet.getString("Nome_Disc")%></option>    <% }
stmt.close();
banco.close(); %>
</select>
...
```

A consulta para obter os dados das disciplinas é realizada através dos comandos `<%=ResultSet.getString(Id_Disc)%>`, fazendo a pesquisa no banco e enviando para o dispositivo WAP.

A Tela com link para selecionar a unidade para consulta das notas é vista na Figura 10.

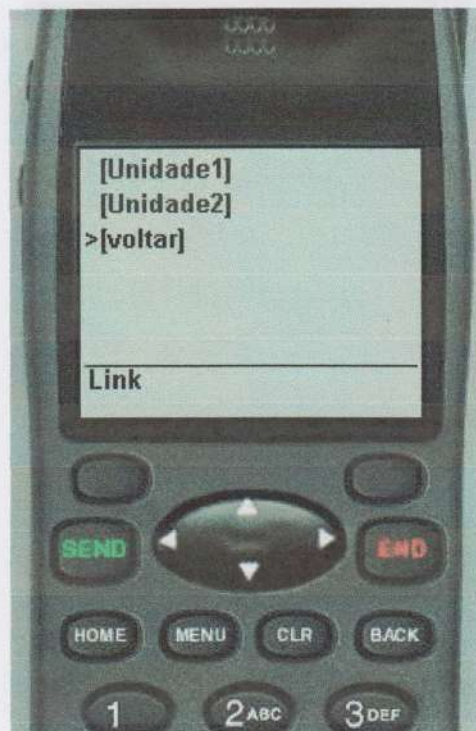


Figura 10: Tela para seleção da unidade

Na tela para escolha da unidade são apresentados dois *links* para que o usuário escolha qual unidade deseja consultar as notas.

Quadro 6: Parte do código WML da página para seleção das unidades

```
</p>
</card>
<card id="carta2" title="MULTIMIDIA">
<p>
<a href="#Un1Mul">Unidade1</a><br/>
<a href="#Un2Mul">Unidade2</a><br/>
<a href="#carta1">voltar</a>
```

Após a seleção da unidade, a próxima carta é acessada e então o usuário pode consultar suas notas, o acesso é feito pela tag `<href>` que faz referencia a outra pagina que esta dentro do mesmo *deck*.

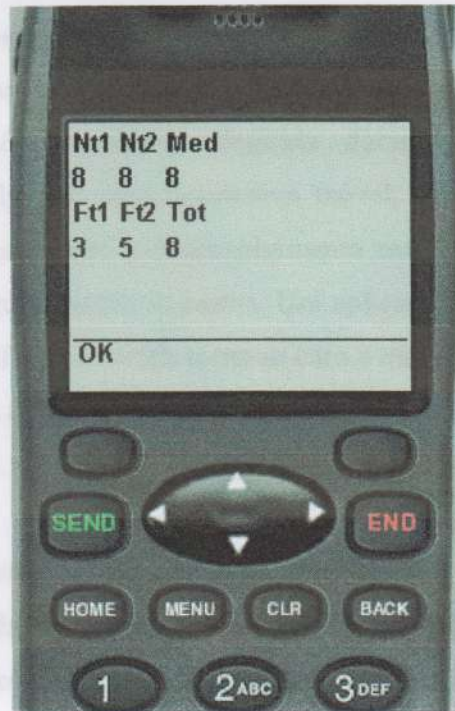


Figura 11: Tela de consulta das notas e faltas

Quadro 7: Código WML com resultado da consulta de notas e faltas

```
<card id="Multi" title="Exemplo 3"><p><small>
<table columns="2" align="center">
<tr><td>Nt1</td>
<td>Nt2</td>
<td>Med</td>
</tr><tr>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
```

O código WML acima apresenta em forma de tabela usando a tag `<td>`, o resultado da consulta após o usuário ter escolhido o *link* referente à unidade que deseja consultar suas notas.

4 CONCLUSÃO

Com o desenvolvimento do protótipo do aplicativo WAP para acessar as notas dos alunos por meio de um dispositivo móvel, foi possível identificar os problemas relacionados com o desenvolvimento de aplicações que utilizam a tecnologia sem fio, desde a fase de análise até o projeto da interface do sistema.

A partir da experiência teórica adquirida durante a pesquisa e a prática no desenvolvimento de uma aplicação para dispositivo móvel, constatamos que, normalmente, as limitações destes dispositivos tornam o desenvolvimento mais específico e, conseqüentemente, mais lento, elevando, em alguns casos, os custos. Um aplicativo usável em um dispositivo com uma interface bastante específica e limitada torna-se caro e mais difícil de desenvolver.

A solução encontrada foi desenvolver aplicações leves, com o mínimo possível de entrada dos usuários, como são as típicas aplicações WAP. O protótipo desenvolvido seguiu essa premissa.

Para reduzir os custos e o tempo de adequação para o uso da tecnologia de desenvolvimento, foi utilizada a linguagem Java com a tecnologia JSP, amplamente conhecida. Ambas são de domínio público e possuem diversos ambientes de desenvolvimento gratuitos.

A usabilidade das interfaces WAP, de acordo com os testes feitos no protótipo é um dos principais pontos de análise durante o projeto do sistema, pois as interfaces do aplicativo podem sofrer uma pequena variação na apresentação de seu conteúdo para o usuário. Isso acontece devido à variedade de browsers acarretando problemas de incompatibilidade.

4.1 Considerações finais

O surgimento da tecnologia *Wireless* e a popularização do acesso às redes de telefonia móvel, levaram a Internet, por meio dos dispositivos móveis, a se tornar uma poderosa ferramenta de acesso aos sistemas de informação. O protocolo WAP apesar de suas limitações e deficiências de segurança, tem boas perspectivas futuras, pois tem uma interessante proposta de disponibilizar serviços diferenciados da chamada Internet móvel, com acesso à informação em qualquer lugar e a qualquer hora.

A telefonia móvel tem avançado bastante nos últimos anos, de acordo com Oliveira (2000) em 1990 quando o telefone celular foi introduzido no Brasil, existiam, segundo a ANATEL (Agencia Nacional de Telecomunicações), apenas 667 aparelhos em operação em todo país. No ano seguinte, o numero passou para 6.700 aparelhos e, apesar de um grande salto, ainda era uma quantidade inexpressiva para um país com 180 milhões de habitantes.

Em 1992, o numero de aparelhos subiu para 30.000 e, desde então, vem crescendo em progressão geométrica. Atualmente segundo dados oficiais, existem um aparelho para cada três habitantes em nosso país, o que remete a marca de cerca de 60 milhões de aparelhos.

Concluimos que o modelo de programação WAP, apesar de suas limitações, abre um mercado com grandes potencialidades. Serviços de acesso a WEB mais eficientes, feitos a partir do desenvolvimento de aplicativos usáveis, são amplamente desejados pela indústria de Telecom, que almeja entrar no mercado da computação móvel.

REFERÊNCIAS BIBLIOGRÁFICAS

- AREHART, Charles et al. **Professional WAP**. United States of América: Editora Wrox press Ltd, 2000. 813p.
- CONALLEN, Jim. **Desenvolvimento de Aplicações Web com UML**. Tradução Altair Dias Caldas de Moraes, Cláudio Bezerra Dias. Rio de Janeiro: Campus, 2003. 476p.
- DEMÉTRIO, Rinaldo. **A Tecnologia WAP: Aprenda a criar sites para celulares com a linguagem WML**. São Paulo: Érica, 2000. 222p.
- FIELDS, Duane K.; KOLB, Mark A.. **Desenvolvendo na Web com Java Serve Pages**. Tradução Rejane Freitas. Rio de Janeiro: Ciência Moderna Ltda, 2000. 559p.
- HORSTMANN, Cay S; CORNELL, Gary. **Core Java 2 volume 1 - Fundamentos**. Tradução Joao Eduardo Nobrega Tortello. São Paulo: Makron Books, 2001, 654p.
- MANN, Steve. **Programando Aplicativos WAP Wireless Application Protocol**. Tradução : Lávio Pareschi. São Paulo : Makron Books, 2001, 232p.
- OLIVEIRA, Wilson J.. **WAP Tecnologia e Segurança**. Florianopolis-SC: Visual Books, 2000. 164p.
- SILBERCHATS, Abraham; KORTH, Henry F.; SUDARSHAN, S; **Sistema de Banco de Dados**. 3ª Edição, São Paulo : Makron Books, 1999. 778p.
- SILVEIRA, Marcelo. **Guia de Consulta Rápida WAP**. São Paulo: Novatec Editora Ltda, 2000. 128p.
- THOMPSON, Maco Aurélio. **Java 2 e Banco de Dados: aprenda na prática a usar Java e SQL para acessar banco de dados relacionais**. São Paulo: Érica, 2002. 198p.