UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

ESCOLA DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELETRÔNICA E DE COMPUTAÇÃO

SISTEMA DE INFORMAÇÕES MÓVEIS PARA DEPARTAMENTO DE COMBUSTÍVEIS

Autora: ______________________________

Ingrid Teixeira

Orientadores: ______________________________

Prof. Aloysio de Castro Pinto Pedroza, Dr.

____

Prof. Antônio Cláudio Gómez de Sousa, M.Sc.

Examinadores: ______________________________

Prof. José Ferreira de Rezende, Dr.

____

Prof. Marcelo Luiz Drumond Lanza, M.Sc.

DEL

SETEMBRO 2002

Ao meu avô João Baptista Martins, meu amigo. Será sempre para mim um referencial de bom-humor perante às dificuldades da vida, compreensão de que cada um de nós é uma individualidade e que como tal possui necessidades distintas, carinho com o próximo e fé em Deus.

A Deus, nosso Pai Eterno.

À minha mãe Eloisa Elena Martins Teixeira, com sua sensibilidade, dedicação e orientação e amor me apoiou em todos os momentos de minha vida. Mais que uma professora de matemática, uma educadora preocupada em incentivar o melhor que existe em cada um de nós.

Ao meu pai Ademar Teixeira, que , do seu jeito, sempre esteve ao lado da família nos apoiando e incentivando.

À minha irmã e amiga Illene Teixeira, companheira que com suas palavras sinceras e corretas sempre me auxiliou. Exemplo de firmeza e determinação nos estudos e na vida.

A todos meus amigos, em especial Carla Faria Gemmal, Cláudia Maria de Freitas da Silva, Lilian Amendola Rendano, Lisiane Bastos Velloso e Luiza Valente Carneiro, pessoas que me cultivaram ao longo do tempo e em diversos momentos de minha vida. Pessoas sempre presentes, embora estando às vezes distante fisicamente.

Ao meu namorado e amigo, Pedro da Fonseca Vieira, que possui um tesouro em seu coração, que me cativa a cada instante.

Aos professores que tive na Universidade Federal do Rio de Janeiro, que enfrentaram dignamente todas as dificuldades presentes no ensino brasileiro e me ajudaram a concluir essa etapa de estudos.

A professora Ivone Alves Rangel do Instituto de Matemática, por ter sido muito mais que professora, sendo para mim uma amiga que marcou para sempre a minha caminhada.

Aos funcionários do Departamento de Engenharia Eletrônica que através de seus trabalhos proporcionaram um ambiente capaz de permitir o meu desenvolvimento.

Aos professores Aloysio de Castro Pinto Pedroza e Antonio Cláudio Gómez de Souza, orientadores desse trabalho, pela paciência, bom-humor e dedicação durante as etapas de desenvolvimento do meu projeto.

Aos professores avaliadores deste trabalho José Ferreira de Rezende e Marcelo Luiz Drumond Lanza, exemplos de professores dedicados aos alunos e empenhados na arte de ensinar.

O *Sistema de Informações Móveis para Departamento de Combustíveis* permite a integração de serviços e informações necessárias para o suporte às atividades de uma empresa de combustíveis que tenha funcionários trabalhando fora dos seus escritórios, junto aos clientes.

Os serviços incluem a funcionalidade necessária para permitir o cadastro e consultas inteligentes sobre os preços encontrados em postos de gasolina da empresa e de sua concorrência. Além disso, suporta funções de cadastro e envio de mensagens para os usuários do sistema. As informações sobre os preços são classificadas de acordo com regras pré-definidas, podendo estar disponíveis para um grupo de usuários destinados através do *Web browser* do aparelho celular.

O Sistema de Informações Móveis para Departamento de Combustíveis está inserido no contexto das aplicações para telefones celulares, representando um exemplo prático de oferta de serviços através da utilização da infra-estrutura de telefonia celular existente hoje na cidade do Rio de Janeiro.

FMMIS

TCP/IP

WAP

Orientação a Objetos

Redes de Computadores Móveis

MySQL

PHP

Modelagem de Sistemas com UML

Dedicatória ii

Agradecimento iii

Resumo v

Palavras-Chave vi

Índice vii

Índice de Figuras x

CAPÍTULO 1 - Introdução 12

CAPÍTULO 2 - Embasamento teórico 14

2.1. WML 14

2.2. PHP 16

2.3. Conceitos Básicos de Orientação a Objetos 17

2.4. UML 17

CAPÍTULO 3 - Arquitetura WAP 19

3.1. Limitações dos Dispositivos e do Ambiente Wireless 19

3.2. O Modelo da World-Wide Web 21

3.3. O Modelo WAP 23

3.4. Melhoramento de Performance – WAP Proxy 25

3.5. Os Elementos da Rede WAP 26

3.6. A Arquitetura do Cliente WAP 29

3.7. Componentes da Arquitetura WAP ____ 30
3.7.1. Rede de Suporte ____ 31
3.7.2. Serviços de Transporte ____ 31
3.7.3. Serviços de Transferência ____ 32
3.7.4. Serviços de Sessão ____ 32
3.7.5. Estrutura da Aplicação ____ 33
3.7.6. Serviços de Segurança ____ 34
3.7.7. Serviços de Descoberta ____ 35
3.8. Configurações WAP ____ 35
CAPÍTULO 4 - Modelagem do Sistema ____ 38
4.1. Casos de Uso ____ 38
4.2. Modelo de Classe ____ 41
4.3. Diagramas de Interação ____ 43
CAPÍTULO 5 - Implementação ____ 44
5.1. Banco de Dados ____ 44
5.1.1. Criação e Carga Iniciais do Banco de Dados ____ 44
5.1.2. Tabelas e seus Relacionamentos ____ 48
5.2. PHP ____ 50
5.2.1. Telas do Sistema ____ 52
5.2.2. Tela Inicial do Sistema ____ 52
5.2.3. Tela de Abertura ____ 55
5.2.4. Tela de Consultas ____ 56
5.2.5. Tela de Escolha da FAZ ____ 56
5.2.6. Tela de Escolha do Posto ____ 57

5.2.7. Tela com o resultado final da Consulta Por Preços de Produtos e Status das Solicitações _______ 57
5.2.8. Tela de Cadastros _______ 58

CAPÍTULO 6 - Simulações e Resultados Obtidos _______ 59

6.1. Cadastro de pedidos de descontos _______ 59

6.2. Consulta dos pedidos feitos – Perfil GT _______ 61

6.3. Cadastro de mensagens _______ 62

CAPÍTULO 7 - Conclusão _______ 64

7.1. Continuações e melhorias a serem agregadas ao FMMIS _______ 64
7.1.1. Criptografia para a troca de mensagens com o servidor _______ 64

Bibliografia _______ 65

Apêndice 1 – Acrônimos utilizados _______ 68

Apêndice 2 – Especificação de requisitos de software _______ 69

Apêndice 3 – Plano de Projeto para Software _______ 85

Apêndice 4 – CD-Rom _______ 94

FIGURA 2-1EXEMPLO DE WML 15

FIGURA 2-2 EXEMPLO DE SINTAXE PHP 16

FIGURA 3-1 O MODELO DE PROGRAMAÇÃO *WWW* 22

FIGURA 3-2 O MODELO DE PROGRAMAÇÃO WAP 23

FIGURA 3-3 WAP PROXY 25

FIGURA 3-4 EXEMPLO DE REDE WAP 26

FIGURA 3-5 PROXY NO CLIENTE MÓVEL 27

FIGURA 3-6 PROXY DO CLIENTE E DA REDE INTERAGINDO 27

FIGURA 3-7 PROXY DO CLIENTE E DA REDE UTILIZANDO PROTOCOLOS ESPECIAIS DE TRANSFERÊNCIA 28

FIGURA 3-8 ARQUITETURA DO CLIENTE WAP 29

FIGURA 3-9 ARQUITETURA DO PROTOCOLO 30

FIGURA 3-10 CONFIGURAÇÃO ORIGINAL WAP 35

FIGURA 3-11 CONFIGURAÇÃO WAP COM PROXY 36

FIGURA 3-12 CONFIGURAÇÃO WAP ACESSANDO DIRETO O SERVIDOR WEB 37

FIGURA 4-1 – CASOS DE USO DO SISTEMA 39

FIGURA 4-2 – CLASSES DO FMMIS 41

FIGURA 5-1– TABELA COM OS RELACIONAMENTOS DO BANCO FMMIS 49

FIGURA 5-2 – TELA INICIAL FMMIS 52

FIGURA 5-3 – TELA DE ABERTURA DO FMMIS 55

FIGURA 5-4 – TELA DE CONSULTAS DO FMMIS 56

FIGURA 5-5 – TELA DE ESCOLHA DA FAZ 56

FIGURA 5-6 – TELA DE ESCOLHA DO POSTO 57

FIGURA 5-7 – TELA FINAL DA CONSULTA POR PREÇOS DOS PRODUTOS 57

FIGURA 5-8 – TELA FINAL DA CONSULTA DO STATUS DAS SOLICITAÇÕES FEITAS ________ 58
FIGURA 5-9 – TELA DE CADASTROS DE PEDIDOS DE DESCONTOS, PERFIL GT ___________ 58
FIGURA 5-10 – TELA DE ESCOLHA DO PRODUTO___________________________________ 58

Muitas empresas possuem, além dos escritórios centrais, funcionários que realizam suas atividades junto aos clientes diretamente. E para tanto, precisam obter informações pertinentes à realização de suas tarefas de modo satisfatório.

O objetivo do Sistema de Informações Móveis para Departamento de Combustíveis é criar um ambiente que permita consultas aos preços vigentes em determinadas áreas de vendas e realizar cadastros de pedidos diretamente. Além disso, o sistema permitirá futuras expansões.

A motivação deste sistema vai além de atender ao Departamento de Combustíveis propriamente dito. É um ambiente, baseado em banco de dados e em WAP, que permite que outras idéias surjam decorrentes de sua implementação.

Ademais, todo o projeto foi desenvolvido de maneira modular e com todos os comentários em todos os níveis, com a finalidade de permitir o reuso e a modificação rápida e simples a *posteriori*.

A presente documentação está dividida de maneira a permitir não só o entendimento do funcionamento do sistema, mas também de maneira a reproduzir o modo como este foi concebido, habilitando o leitor também a dele se utilizar de maneira simples e direta.

O Capítulo 2 oferece os subsídios necessários ao entendimento do desenvolvimento do projeto. Esse embasamento abrange as linguagens para a construção da interface com o usuário - WML (*Wireless Markup Language*) e PHP (*Pre Hypertext Processor*), e a modelagem do projeto utilizando os conceitos de orientação a objetos e UML.

O Capítulo 3 é um estudo feito sobre a arquitetura WAP. Esse capítulo descreve a infra-estrutura para o ambiente móvel celular, abrangendo os protocolos de comunicação existentes nos aparelhos celulares e na rede de suporte e os modos de operação da mesma.

O Capítulo 4 é destinado à modelagem do sistema através dos conceitos de orientação a objetos. Toda a modelagem do projeto passou pelas fases de especificação e plano de projeto de *software*. Além disso, foi utilizada a UML como a linguagem para essa modelagem.

O capítulo 5 destina-se à implementação do projeto. Criação da interface com o usuário e construção da base de dados utilizada. Neste capítulo são mostradas todas as telas presentes no sistema desenvolvido através das quais o usuário tem acesso às funcionalidades do mesmo.

O capítulo 6 permite que o leitor visualize como se processa a interação do sistema com o usuário, mostrando alguns casos de uso do mesmo. São exibidas algumas seqüências de telas que o usuário deve percorrer a fim de verificar uma determinada funcionalidade presente no sistema.

O capítulo 7 é destinado às conclusões a respeito do projeto, enfocando de maneira especial as futuras melhorias e expansões.

O processo de desenvolvimento do projeto pode ser dividido em 3 grandes vertentes: a apresentação da Infra-Estrutura utilizada, a Modelagem do sistema através da UML e a Implementação do Banco de Dados (feito em *MySQL*) e das páginas visualizadas pelo usuário, feitas em PHP.

Cada uma das etapas, para finalidade didática, será tratada em uma sessão à parte, com exemplos de códigos e usos, com o objetivo de fornecer um entendimento sobre os mecanismos utilizados para implementar o sistema como um todo.

Nas várias etapas do desenvolvimento do projeto, fez-se, naturalmente, o estudo aprofundado de técnicas e fundamentos que tornassem viáveis a sua implementação. A seguir, apresenta-se de forma essencial como base para entender o correto funcionamento deste trabalho.

A apresentação do embasamento teórico não visa um treinamento ou conhecimento profundo dos assuntos, apenas a formação da idéia necessária à compreensão do projeto. Para posteriores consultas podem-se consultar a bibliografia, situada ao final deste documento.

2.1.WML

O Wireless Markup Language (*WML*) [WML] é baseado no HTML [Ragget, 1998] da WWW e no HDML [King, 1997]. WML é especificado com um documento do tipo XML [W3C, 1998]. Ao se projetar uma página WML deve-se considerar que os dispositivos celulares possuem, na sua maioria, pequeno *displays*[1], baixa capacidade de processamento, pouca memória e *links* sem-fio com baixa e limitada largura de banda.

WML utiliza o conceito de *deck* e *card*. Um documento WML é composto de um ou mais *cards* , que podem estar agrupados em um *deck*. Um deck WML é similar a uma página HTML, no que diz respeito a ser especificada pela URL e ser a unidade de transmissão do conteúdo.

[1] Displays – tela, visor do aparelho em questão.

A figura 2-1 é um exemplo de sintaxe WML. Após a palavra WML o primeiro CARD é definido. O primeiro card do deck exibe[2] um texto depois de carregado. O elemento DO faz com que o agente usuário exiba o segundo card. Nesse segundo card pode-se escolher[2] dentre as opções exibidas.

```
<WML>
        <CARD>
        <DO TYPE = "ACCEPT">
                <GO URL= "#card_two"/>
        </DO>
        EXEMPLO DE UM CARD SIMPLES!
        </CARD>
        <CARD NAME= "card_two">
        <SELECT KEY= "PIZZA">
                <OPTION VALUE= "Mar">
                        Margherita</OPTION>
                <OPTION VALUE= "Fun">
                        Funghi</OPTION>
        </SELECT>
        </CARD>
</WML>
```

Figura 2-1Exemplo de WML

WML pode ser codificado utilizando uma representação binária compacta para economizar largura de banda em um enlace sem-fio. Esta representação compacta é baseada no formato de conteúdo XML binário. A codificação binária do WML é somente uma versão especial, sendo válida para qualquer linguagem baseada em XML. O formato compacto permite a transmissão sem a perda da funcionalidade ou da informação. Por exemplo, o prefixo de URL http://, seria codificado como 4B. Esses códigos em bytes são muito mais eficientes do que o HTML puro.

[2] Exibe – neste caso a exibição pode ser, além de texto, uma voz de saída. Assim como as escolhas dos itens pode ser feita através de voz de entrada.

2.2.PHP

PHP é uma linguagem de *script* do lado do servidor (SSS), específica para acesso e processamento de Linguagens Markup como HTML ou WML.

A maioria de sua sintaxe é trazida do C, Java e do Perl com algumas funcionalidades específicas dele. O objetivo da linguagem é permitir a programação de conteúdo WML de maneira dinâmica e rápida, com uma linguagem poderosa.

PHP possui capacidades de conexão direta com bancos de dados com instruções específicas para tanto. Possui também capacidades intrínsecas de criptografia (implementadas por biblioteca especial) e tratamento de imagens e animações (que fogem ao escopo deste projeto).

A sintaxe de PHP é embutida no WML por um rótulo especial <? ?>, conforme podemos ver na figura 2-2:

```
<WML>
        <CARD>
        <?
                    Oi, sou um script PHP!
        ?>
        </CARD>
</WML>
```

Figura 2-2 Exemplo de sintaxe PHP

Para o cliente a parte <? ?> é invisível, porque, conforme foi dito, é processada do lado do servidor, e o cliente não tem qualquer acesso a mesma. Quando um cliente solicita uma página com conteúdo em PHP, o servidor Web pré-processa essa página. Isso significa que o servidor da Web percorre a página de cima para baixo, procurando por seções do PHP, as quais tentará resolver. O analisador de sintaxe identificará todas as variáveis atribuídas (marcadas pelo sinal de cifrão *$*) e tentará conecta-las a comandos do PHP posteriores. O pré-processador por fim retornará uma página HTML normal para o navegador do cliente. Os conteúdos são, deste modo, dinamicamente exibido conforme as iterações são feitas.

2.3. Conceitos Básicos de Orientação a Objetos

Uma *classe* é um modelo utilizado para criar objetos específicos e relacionados, uma descrição de um conjunto de instâncias que compartilham as mesmas operações, atributos, relacionamentos e semântica. Um *objeto* real é uma instância de uma classe, possui dados (*atributos*) e aguarda por executar serviços (*métodos*). Uma classe define os atributos compartilhados por todos os objetos criados a partir dela e estabelece *relacionamentos* com outras classes: duas classes podem compartilhar alguns atributos mas manipulam outros atributos diferentemente.

Na programação orientada a objetos, a *herança* permite expressar relacionamentos entre classes onde as classes base derivam de uma subclasse mais complexa. Uma classe básica contém todos os atributos que são compartilhados entre as classes derivadas da classe base.

2.4. UML

Modelar um sistema complexo é uma tarefa extensiva sendo necessária a descrição de vários aspectos diferentes incluindo o funcional (estrutura estática e interação dinâmica), não funcional (tempo de processamento, confiabilidade, produção) e organizacional (organização do trabalho, mapeamento e código). Cada visão é descrita em um número de diagramas que contém informações enfatizando um aspecto particular do sistema.

A *UML – the Unified Modeling Language* é uma linguagem de modelagem de objetos unificada. A notação da UML é uma combinação de sintaxe vinda de várias fontes, mas com vários símbolos removidos das técnicas anteriores e com alguns símbolos novos agregados para tratar situações específicas.

Na sessão sobre a modelagem do sistema, apresenta-se a UML através do desenvolvimento do projeto em si.

Com o crescimento da Internet, suas aplicações e as comunicações móveis, muitas soluções proprietárias foram surgindo com o intuito de promover o acesso aos serviços da Internet pelos dispositivos móveis e sem fio. Essas soluções buscavam resolver os principais problemas que aplicativos *web* encontravam quando utilizados em um ambiente sem fio e móvel. Basicamente, as idéias iniciais se concentravam em prover melhorias ao HTML (*Hyper Text Markup Language*), suporte a diferentes arquiteturas de sistemas, e protocolos de transferência adaptados aos requisitos do acesso móveis.

A fim de se evitar muitas ilhas isoladas de soluções incompatíveis, ex: soluções para GSM, IS-136, ou para alguns fabricantes, o *Wireless Application Protocol Forum (WAPForum)* foi fundado em junho 1997 pelas empresas Ericsson, Motorola, Nokia, e Unwired Planet (WAPForum, 1998a).

Os objetivos básicos do *WAPForum* são de trazer para os celulares digitais e outros terminais móveis e sem fio (ex PDA, laptops) o conteúdo da Internet e outros serviços de dados. Além disso, deseja-se dispor de um conjunto de protocolos capazes de permitir comunicação sem fio através de várias tecnologias distintas, como GSM, CDPD, UMTS etc.

3.1. LIMITAÇÕES DOS DISPOSITIVOS E DO AMBIENTE WIRELESS

WAP encontra-se na convergência de dados móveis, telefonia e Internet. Acontece, porém, que a maioria da tecnologia original desenvolvida para a Internet visava atender aos *PCs* (Personal Computers) e grandes computadores, uma largura de banda de média a grande e, geralmente, uma rede de dados mais confiável (WAP Forum, 2001).

Os *dispositivos* portáteis e móveis apresentam um ambiente computacional mais "restrito" quando comparado aos PCs. Esses dispositivos têm limitações de potência causada pela tecnologia atual de baterias. Isto implica:

CPUs *(Central Processor Units)* menos velozes,

Menos capacidade de memória RAM *(Random Access Memory)* e ROM *(Read-Only Memory)*,

Consumo de potência restrito,

Display pequeno, e

Dispositivos de entrada de dados diferentes (ex. um teclado para telefone, entrada de dados por voz).

Além disso, as redes móveis de dados apresentam um *ambiente* de comunicação mais restrito quando comparado às redes fixas. Devido a limitações fundamentais de potência, espectro de freqüência disponível, e mobilidade, redes sem fio tendem a ter:

Menor largura de banda,

- o Com o aumento da largura de banda, aumenta o consumo de potência. Nos dispositivos móveis causa redução da vida útil da bateria.

Maior tempo de latência,

- o O ambiente móvel é caracterizado por uma grande variedade de latência de rede, podendo alcançar dezenas de segundos. Soma-se a isto, o fato da latência poder variar, dependendo das características da transmissão e da carga da rede em uma area. O tempo de latência também é aumentado pelo roteamento, pelos algoritmos de correção de erro e as características para se evitar congestionamento de uma rede particular.

Menor estabilidade de conexão,

- o Nas redes móveis, acontecem, por exemplo, atenuações dos sinais de rádio por longas distancias (Schiller, 2000), regiões de sombra ou bloqueio do sinal, propagação *multipath*, causando atrasos e etc

Disponibilidade menos previsível

- o As redes móveis necessitam de um planejamento de freqüência para evitar interferência, complexa infraestrutura , como antenas e *switches* para estabelecer, registrar e encaminhar as chamadas.

As redes móveis estão crescendo em complexidade e os custos de todos os aspectos para prover serviços com valores agregados também estão crescendo. A fim de se atender aos operadores de redes móveis, as soluções devem ser:

Interoperáveis – terminais de diferentes fabricantes podem se comunicar com os serviços das redes móveis,

Escaláveis – operadores de redes móveis são capazes de escalar os serviços para atender aos clientes,

Eficientes – prover qualidade do serviço *(QoS – Quality of Service)* mais adequada ao comportamento e característica da rede móvel,

Confiáveis – prover uma plataforma consistente para suporte dos serviços, e

Seguras – possibilitar que os serviços se estendam por redes móveis potencialmente inseguras, preservando a integridade dos dados do usuário: protegendo os dispositivos e serviços de problemas como a perda da confidenciabilidade.

3.2. O Modelo da World-Wide Web

A arquitetura *World-Wide Web (WWW)* dispõe de um modelo poderoso e muito flexível de programação, ilustrado na figura abaixo [WAPForum, 2001]. Tanto a aplicação quanto o conteúdo possuem padrões para o formato dos dados, e são acessadas por aplicativos denominados *web browsers*. Os *web browsers* são aplicativos de rede, isto é, eles enviam pedidos para determinados servidores da rede e o servidor responde com os dados padronizados.

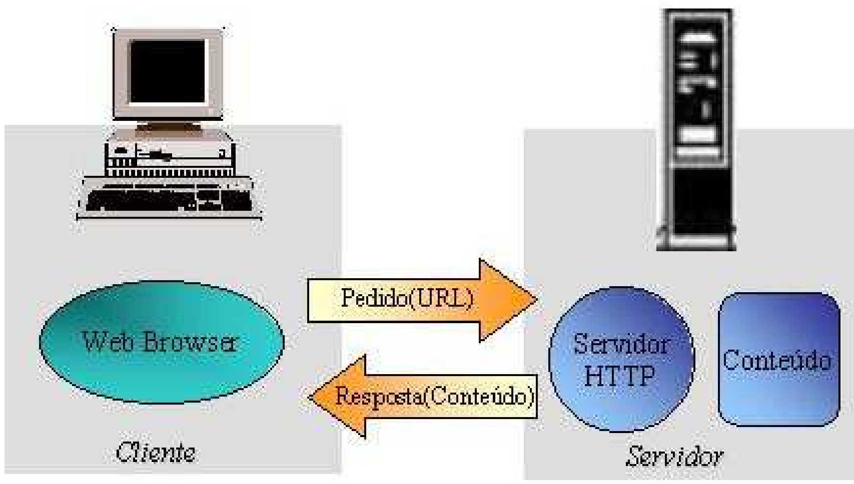


Figura 3-1 O Modelo de Programação *WWW*

Os padrões da *WWW* especificam muitos mecanismos necessários para se construir um ambiente, incluindo:

Nome Padrão

- Todos os servidores da WWW são nomeados pelo padrão Internet *Uniform Resource Locator (URL)* [RFC2396].

Tipo do Conteúdo

- Todo o conteúdo na WWW é de um tipo específico, permitindo aos *web browsers* processá-lo corretamente [RFC2045, RFC2048].

Formato do Conteúdo Padrão

- Todos os *web browsers* suportam um conjunto de padrões para o conteúdo. Eles incluem o HyperText Markup Language (HTML) [HTML4], linguagens script e outros formatos.

Protocolos Padronizados

o O protocolo mais utilizado na WWW é o HyperText Transport Protocol (HTTP) [RFC2616], operando em cima dos protocolos TCP/IP [STD0007].

Esta infra-estrutura padronizada permite aos usuários utilizar um grande número de aplicações e serviços oferecidos na Internet.

3.3. O MODELO WAP

O modelo de programação WAP mostrado na figura 3-2 é semelhante ao modelo da WWW com algumas modificações. Manter essa certa semelhança produz alguns benefícios para a comunidade de desenvolvedores, como ter um ambiente de programação familiar, utilizar uma arquitetura disponível e a possibilidade de utilizar as ferramentas existentes, exemplo os servidores *web*, ferramentas XML, etc. Otimizações e extensões foram feitas a fim de compatibilizar com o ambiente sem fio. E quando possível, padrões existentes foram adotados ou usados como pontos de partida para a tecnologia WAP [WAPForum, 2001].

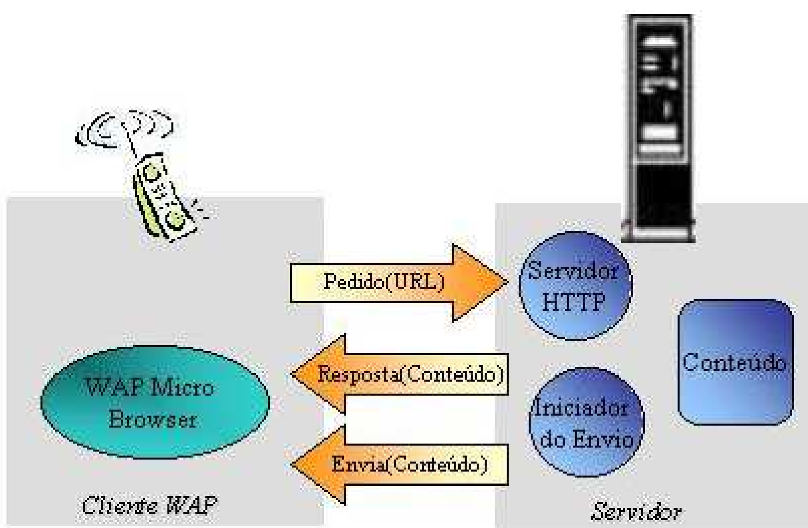


Figura 3-2 O Modelo de Programação WAP

O mecanismo clássico de pergunta-resposta é comumente conhecido como *pull (puxa)* em contraste com o mecanismo *push (empurra)*.

O conteúdo WAP e os aplicativos são especificados em uma série de formatos baseados nos formatos padrões da WWW. O conteúdo é transportado usando-se protocolos padrões de comunicação baseado nos protocolos de comunicação da WWW. O *microbrowser* do WAP no terminal sem fio possui a interface com o usuário e é análogo ao browser da Web.

O WAP define um conjunto de componentes padrões, que permitem a conexão entre os terminais móveis e os servidores da rede, incluindo:

Modelo de endereçamento padrão – WWW

- o URL padrões são utilizadas para identificar os servidores e serviços locais;

Tipo de Conteúdo

- o Permite que o agente usuário WAP processe corretamente as informações recebidas;

Formato do conteúdo padrão

- o Os formatos do WAP são baseados na tecnologia WWW e inclui o markup display, linguagem script e etc;

Protocolos de comunicação padrão

- o Os protocolos de comunicação do WAP permitem a comunicação dos pedidos do browser do terminal móvel ao servidor da web da rede fixa.

3.4.Melhoramento de Performance – WAP Proxy

O WAP utiliza a tecnologia do proxy a fim de otimizar e aperfeiçoar a conexão entre o domínio wireless e o WWW. O proxy WAP pode prover uma variedade de funções, incluindo:

Protocolos do Gateway – os protocolos de gateway traduzem os pedidos feitos pelos protocolos wireless para os protocolos WWW (HTTP e TCP/IP). O gateway também executa a busca DNS dos servidores indicados pelos clientes nos pedidos de URLs;

Codificação e Decodificação do conteúdo – é feito a fim de reduzir o conteúdo em um tamanho mais apropriado para o ambiente sem fio;

Gerenciamento do Perfil do Agente Usuário – descreve as capacidades e as preferências pessoais dos clientes;

Caching Proxy – possibilita uma melhora de performance e utilização da rede, armazenando um *cache* dos recursos mais acessados.

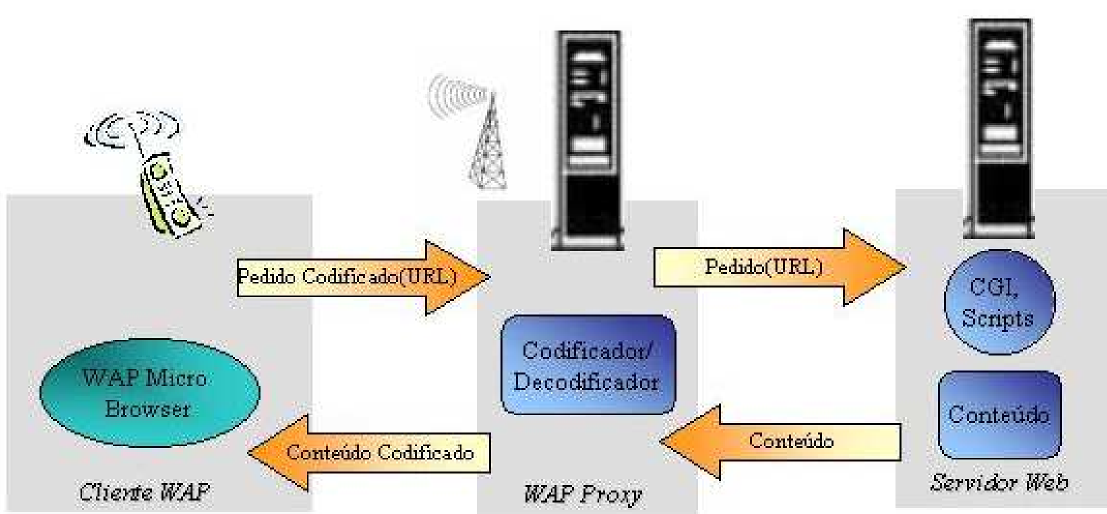


Figura 3-3 WAP Proxy

3.5. *Os Elementos da Rede WAP*

Na figura abaixo temos o desenho de uma típica rede WAP.

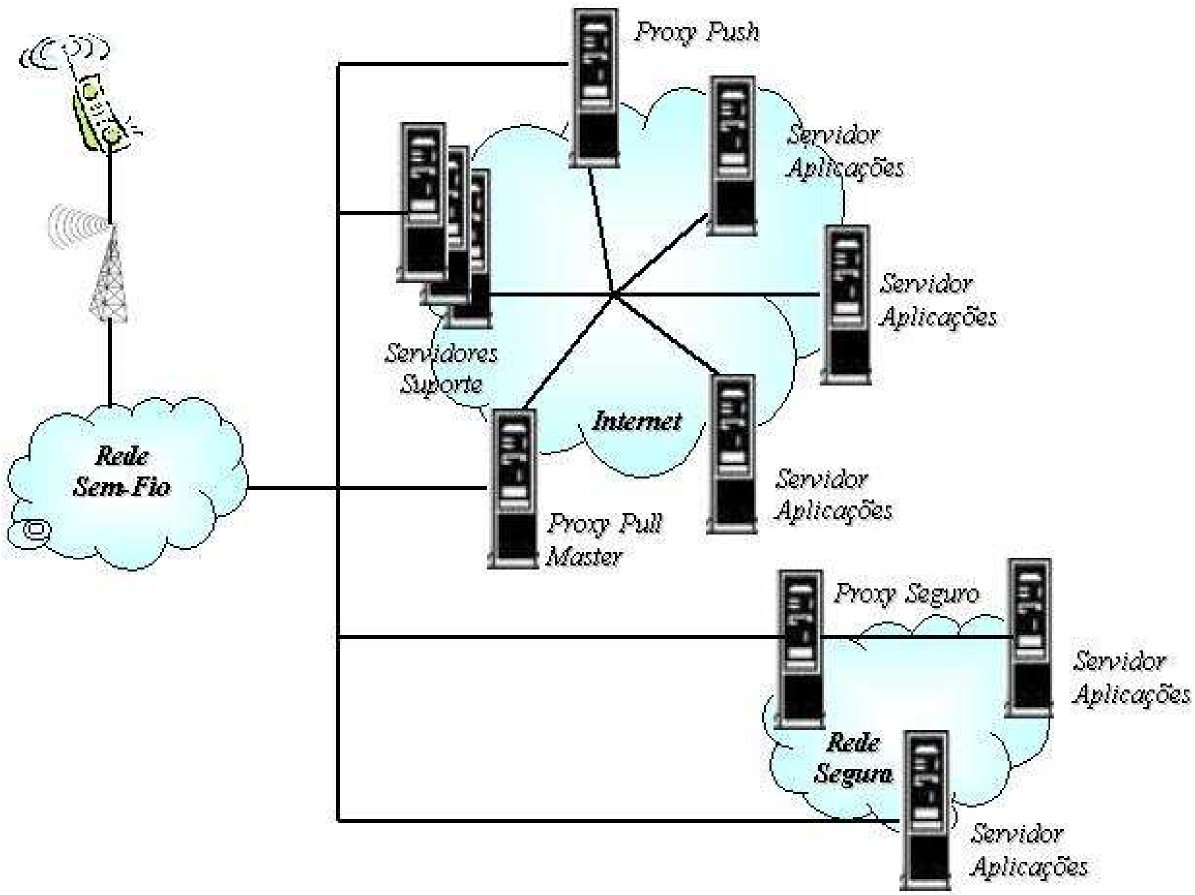


Figura 3-4 Exemplo de rede WAP

Como pode ser observado, o cliente WAP pode se comunicar com os servidores de aplicações por diferentes *proxies* da rede ou diretamente.

Os *proxies* podem estar localizados em vários lugares na arquitetura da rede, como nos provedores de serviço independentes, a fim de auxiliar na melhoria dos serviços prestados pela rede móvel (ex: telefonia, localização e provendo serviços) ou para otimizar a comunicação entre o dispositivo e o servidor de aplicações (ex: "traduzindo" os protocolos e armazenando os “cookies”). Os *proxies* também podem estar localizados em redes seguras a fim de possibilitar um canal seguro entre o dispositivo móvel e a rede.

A fim de possibilitar a navegação na Web pode-se ter um proxy no cliente [Schiller, 2000] como mostrado na figura 5 abaixo. O browser do cliente acessa o servidor Web através do seu proxy, ou seja, o proxy é ao mesmo tempo *servidor* para o browser e *cliente* para o servidor Web.

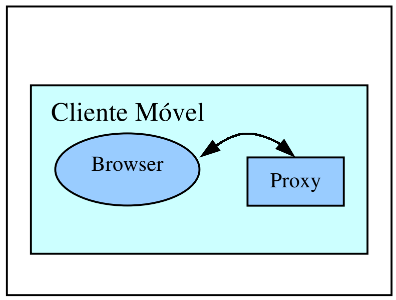


Figura 3-5 Proxy no Cliente Móvel

Os benefícios dos proxies do cliente e da rede podem ser ajustados, resultando em uma arquitetura ilustrada na figura 6. Um exemplo dessa combinação é o WebExpress, encontrado em [Housel, 1996], [Floyd, 1998]. O proxy do cliente pode, informar ao proxy da rede sobre o comportamento do usuário, fazendo com que o proxy da rede pré-selecione páginas de acordo com essa informação.

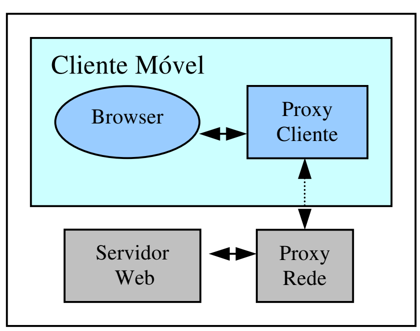


Figura 3-6 Proxy do Cliente e da Rede interagindo

Um subsistema mais “especializado” de rede é mostrado na figura 7. Esta solução tem os mesmos benefícios da mostrada anteriormente, com a diferença de que a transferência do conteúdo pode ser otimizada. Exemplos são a compressão *online* e a substituição de protocolos de transferência como o HTTP e o TCP por protocolos mais adaptados a mobilidade.

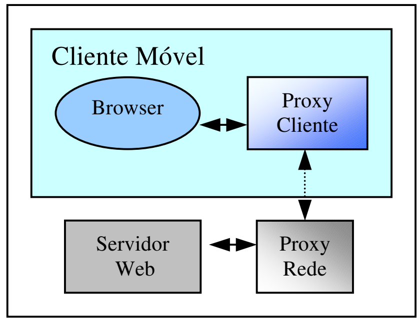


Figura 3-7 Proxy do Cliente e da Rede utilizando Protocolos Especiais de Transferência

Um exemplo dessa arquitetura é o Mowgli [Liljeberg, 1995], [Liljeberg, 1996]. Este sistema possibilita o acesso a Web pelas redes telefônicas de celulares, ou seja, redes com pouca largura de banda e um atraso relativamente maior. Ele não só substitui os protocolos de transporte, mas também implementa modificações no conteúdo esperadas pelos telefones móveis. O browser ainda utiliza o HTTP para o proxy do cliente. O proxy do cliente então utiliza um serviço especializado de transporte, *o serviço de canal de dados Mowgli*, para acessar o proxy da rede. Os proxies do cliente e da rede trocam suas mensagens através das conexões Mowgli *long-lived*, evitando o problema do TCP como o início lento e o fato de existir de uma conexão TCP por pedido HTTP [Schiller, 2000].

3.6. A ARQUITETURA DO CLIENTE WAP

A arquitetura do dispositivo WAP é mostrada na figura 8. A estrutura[3] do aplicativo é tal que possibilita a execução de aplicações WAP, as quais são compostas de *markup*, *script*, conteúdo multimídia, no dispositivo. O modelo de processamento *WAE – WAP Application Environment*. Define a estrutura na qual essas formas de conteúdo interagem.

Os protocolos de rede no cliente WAP são compartilhados entre cliente e servidor. Os *content renderers* interpretam as formas específicas de conteúdo e exibem-na para o usuário final. As funções comuns são aquelas necessárias à estrutura da aplicação, incluindo a persistência e a sincronização de dados.

O *WIM – Wireless Identity Module* contém a identidade do dispositivo e os meios de cifragem[4] que mutuamente autenticam o dispositivo WAP e os servidores, para maiores detalhes, que fogem o escopo do trabalho, consultar [WIM, 2000].

O *EFI – External Functionality Interface* é um mecanismo de se acessar funções externas que estão presentes nos dispositivos móveis.

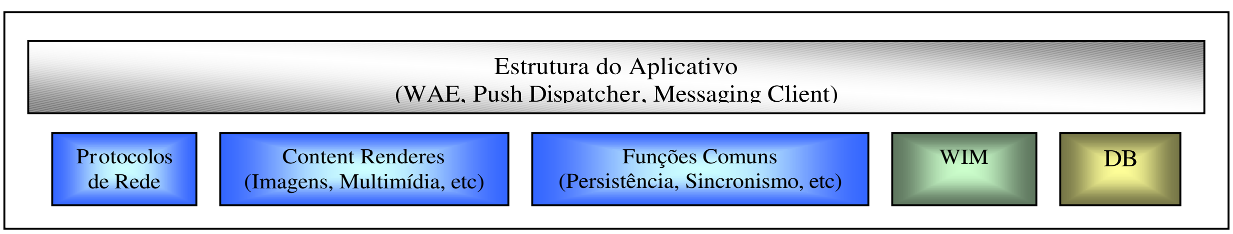


Figura 3-8 Arquitetura do Cliente WAP

---

[3] Do inglês, framework

[4] *Cryptography* – Cifragem, criptografia.

3.7. Componentes da Arquitetura WAP

Uma visão geral da arquitetura WAP é mostrada a seguir na figura 9. Cada camada provê um conjunto de funções e/ou serviços e aplicações a serem utilizados pelas camadas superiores.

Nas próximas seções serão detalhados os principais serviços presentes em cada camada WAP.

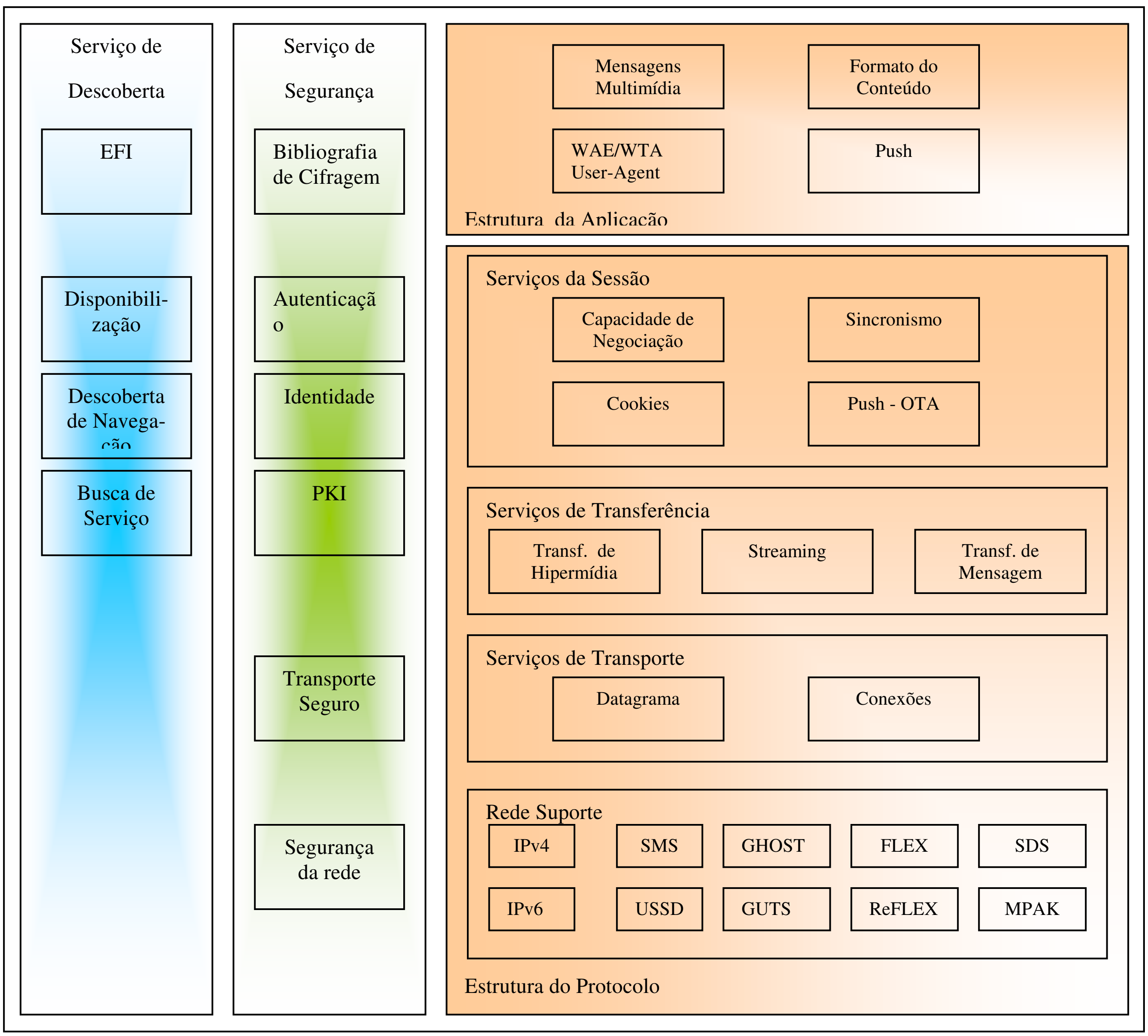


Figura 3-9 Arquitetura do Protocolo

3.7.1. Rede de Suporte[5]

A transmissão de dados utiliza os serviços oferecidos pela rede de suporte, como por exemplo o *GSM – Groupe Spéciale Móbile, Global System for Mobile communications* e o *CDPD – Cellular Digital Packet Data*. WAP não especifica os serviços oferecidos por essa "camada", mas os utiliza e no futuro pretende integrá-los [Schiller, 2000]. Exemplos são serviços de mensagens, como o *SMS – Short Message Service*, dados transmitidos pelos circuitos virtuais, como o *HSCSD – High-Speed Circuit Switched Data* ou dados transmitidos por pacotes, como o *GPRS – General Packet Radio Service*.

Essa "camada" oferece diferentes níveis de qualidade de serviço no que diz respeito à taxa de transmissão, taxa de erro, e atrasos. Os protocolos são projetados para compensar ou tolerar os diferentes níveis de serviço.

Como a camada de Serviço de Transporte provê a interface entre os serviços de suporte (bearer) e o resto do pilar WAP, na especificação da mesma em [WDP], encontram-se os serviços suportados e as técnicas utilizadas nos protocolos de comunicação.

3.7.2. Serviços de Transporte

Através do T-SAP, que é o ponto de acesso através do qual a camada superior se comunica com a camada de transporte, os serviços oferecidos são, dentre outros:

Datagrama – o serviço de datagrama permite o transporte dos dados no qual entidades independentes de dados possuem informações suficientes para que os pacotes sejam roteados da origem até o endereço destino. Os protocolos UDP (*User*

---

[5] Redes de Suporte – do inglês Bearer Networks significa *que serve de suporte, para transportar, etc*. No modelo OSI da ISO, as camadas rede, enlace e física estariam englobadas por essa terminologia. Bearer Networks – infra-estrutura de rede de suporte.

Datagram Protocol) [STD0006] e WDP (*Wireless Datagram Protocol*) [WDP] são utilizados a fim de prestarem este serviço.

Conexão – o serviço de conexão para o transporte é aquele no qual a comunicação se dá em três etapas: o estabelecimento da conexão, transferência de dados em ambos os sentidos de modo confiável e o fechamento da conexão. O protocolo utilizado é o TCP (*Transmission Control Protocol*) [STD0007].

3.7.3. Serviços de Transferência

Os serviços de transferência estruturada de informação entre elementos da rede são por exemplo:

Transferência de Hipermídia – pode ser utilizada a combinação de WSP (*Wireless Session Protocol*) [WSP] e WTP (*Wireless Transaction Protocol*) [RFC2616] para este serviço sobre o serviço de transporte orientado a conexão seguro ou inseguro.

Streaming – permite a transferência de dados como áudio e vídeo.

Transferência de mensagens – transferência assíncrona de mensagens, como de emails ou mensagens instantâneas

3.7.4. Serviços de Sessão

Permite o estabelecimento de um estado compartilhado entre os elementos de rede que podem dividir um pedido de rede ou transferência de dados. Por exemplo a sessão Push permite que o dispositivo WAP esteja pronto e hábil a receber mensagens do proxy Push. Outros serviços são:

Capacidade de negociação – clientes e servidor podem acordar parâmetros de protocolos durante o estabelecimento da sessão. Exemplos de parâmetros são: tamanho do SDU do cliente, pedidos máximos e SDU do servidor.

Push-OTA (*Push Over The Air*) – é o serviço de sessão que permite que mensagens sejam entregues (push) através de uma sessão já existente [PushOTA].

3.7.5. Estrutura da Aplicação

O objetivo é de provê um ambiente de aplicação baseado na World Wide Web (WWW), Internet e tecnologias de Telefonia Móvel. O WAE (*Wireless Application Environment*) não impõe nenhum padrão para a interface homem-máquina, cada fornecedor pode criar a sua própria.

WAE/WTA User-Agent – WAE é um ambiente micro-browser que permite o uso de linguagens markup (WML e XHTML), scripting e serviços de telefonia [WAE].

Push – O serviço de Push provê um mecanismo geral para a rede iniciar a transmissão de dados para as aplicações [PushArchOverview].

Mensagens Multimídia – O Multimedia Message Service (MMS) permite a transferência de mensagens multimídia , como email e mensagens instantâneas para os dispositivos WAP.

Formato do Conteúdo – suporta formato de dados definidos como cor, imagens, áudio, vídeo, animação e etc.

3.7.6. Serviços de Segurança

Os serviços de segurança podem ser encontrados em todas as camadas que formam o WAP. Os serviços oferecidos são de *privacidade*, evita que a comunicação possa ser entendida por partes não destinadas, *autenticação*, estabelece um padrão de comunicação entre as partes, *integridade*, garante que as mensagens são imutáveis e não são corrompidas, *não-repudiação*, impede que as partes neguem o conteúdo das suas mensagens. O PKI é um conjunto de serviços de segurança que permite o uso e a administração de criptografia de chave pública [WPKI], [WAPCert].

3.7.7. Serviços de Descoberta

O serviço de descoberta forma uma parte fundamental na arquitetura WAP e seus serviços podem ser encontrados em muitas camadas. *EFI – The External Functionality Interface* possibilita que as aplicações descubram quais as funções externas/serviços estão disponíveis. O serviço de Lokup permite a descoberta dos parâmetros do serviço, buscando no diretório pelo nome, exemplo: Domain Name System (DNS) [STD0013].

3.8. Configurações WAP

Nas próximas páginas serão mostrados alguns exemplos de configurações possíveis de protocolos para a estrutura de uma rede sem-fio interoperando com uma rede fixa.

Na figura 3-10, tem-se a configuração da arquitetura WAP original. O Gateway WAP converte o serviço de transferência entre os protocolos baseados em datagrama (WSP, WTP, WTLS, WDP) nos protocolos orientados a conexão mais comuns da Internet (HTTP,SSL,TCP).

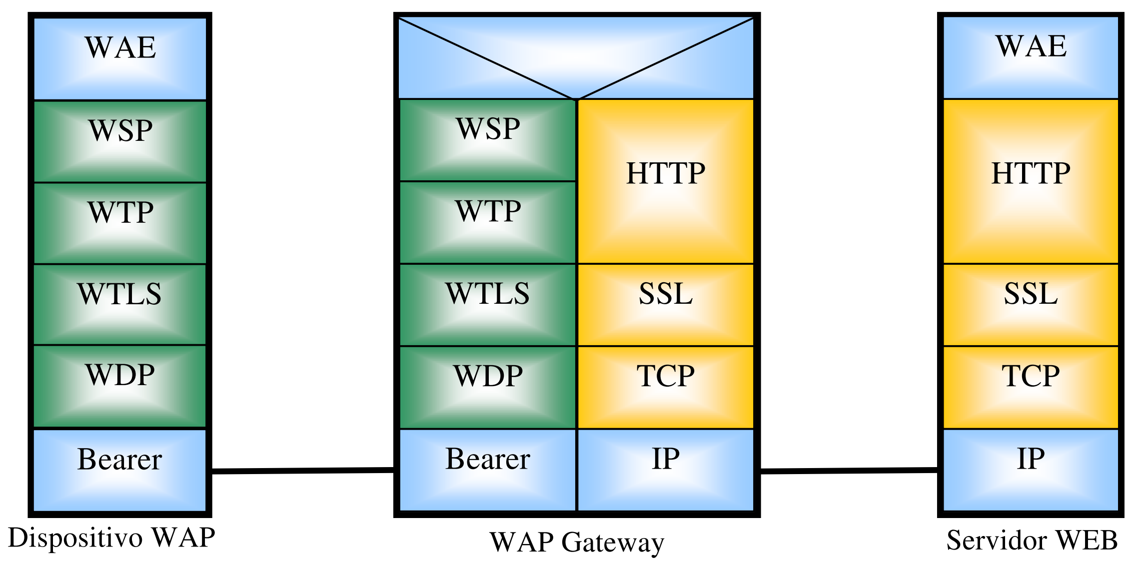


Figura 3-10 Configuração original WAP

Na figura 3-11 a configuração WAP utiliza um proxy HTTP e o dispositivo WAP possui os protocolos TCP e HTTP otimizados para o ambiente sem-fio.

A configuração proxy é largamente utilizada na Internet para permitir o acesso à web, dados multimídia, como música e outros. Esta configuração possui o WAP proxy entre as redes sem-fio e “com fio” e a fim de melhorar a performance utiliza os protocolos TCP e HTTP otimizados, que são compatíveis com o TCP e HTTP presentes na Internet.

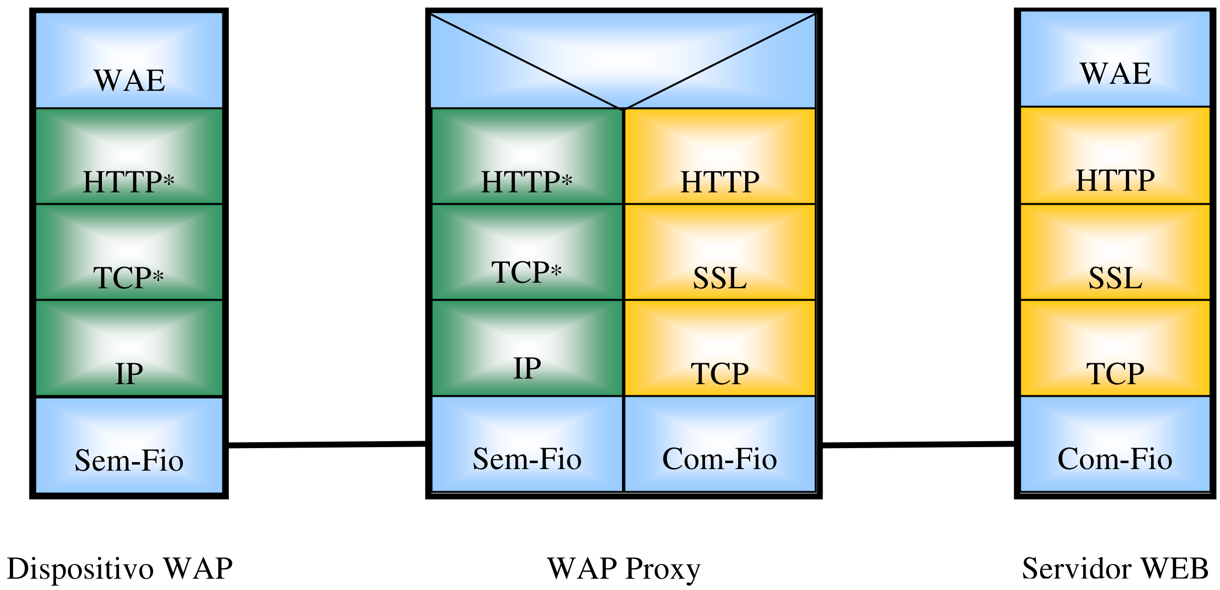


Figura 3-11 Configuração WAP com Proxy

Na figura 3-12, tem-se o dispositivo WAP acessando diretamente um servidor Web via Internet. O roteador wireless IP é parte padrão de uma rede IP e é utilizado entre uma camada (exemplo a camada de enlace sem-fio) e outra (exemplo a camada de enlace da rede com fio). Geralmente essa configuração é utilizada quando se utiliza segurança no enlace como o IPSec.

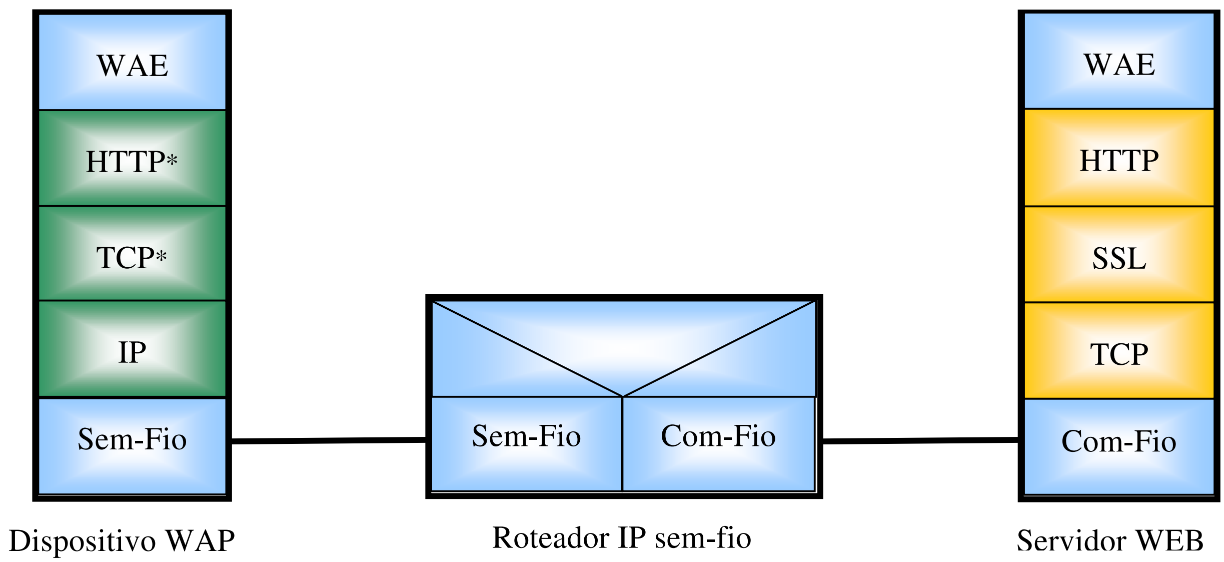


Figura 3-12 Configuração WAP acessando direto o servidor Web

O sistema FMMIS é um sistema de suporte às vendas. O FMMIS tem como objetivo fornecer informações unificadas aos revendedores, incluindo consultas aos preços dos produtos comercializados em sua área de atuação, situações dos pedidos de descontos solicitados, entre outras. Esse sistema permite também que o supervisor cadastre mensagens para seus subordinados, além de possibilitar que sejam feitas solicitações de desconto na venda dos produtos.

A força de vendas está estruturada em filiais, áreas e zonas. Uma filial é responsável pela cobertura de uma área geográfica, estando subdividida em zonas de venda. Cada zona de venda é de responsabilidade de um revendedor titular, que recebe o nome de *gerente de território - GT* e a filial é de responsabilidade de um *gerente de área - GA*.

A solução proposta é descrita através de casos de uso e detalhada em um modelo de desenho que descreve a solução técnica proposta.

4.1. Casos de Uso

O primeiro passo na análise é definir os casos de uso que descrevam o que o sistema FMMIS irá oferecer em termos de funcionalidade. Os atores envolvidos são humanos (os gerentes de território e de área).

Os casos de uso previstos são:

- Recebe mensagens sobre segurança
- Consulta produtos dos postos da cia e da concorrência dentro da FAZ
- Consulta o status da solicitação feita (individual)
- Consulta avisos da cia
- Consulta avisos do GA

- Efetua pedidos de descontos nos preços dos produtos (perfil GT)
- Consulta o status das solicitações feitas por cada GT subordinado
- Efetua pedidos de descontos nos preços dos produtos (perfil GA)
- Cadastra avisos para os GT

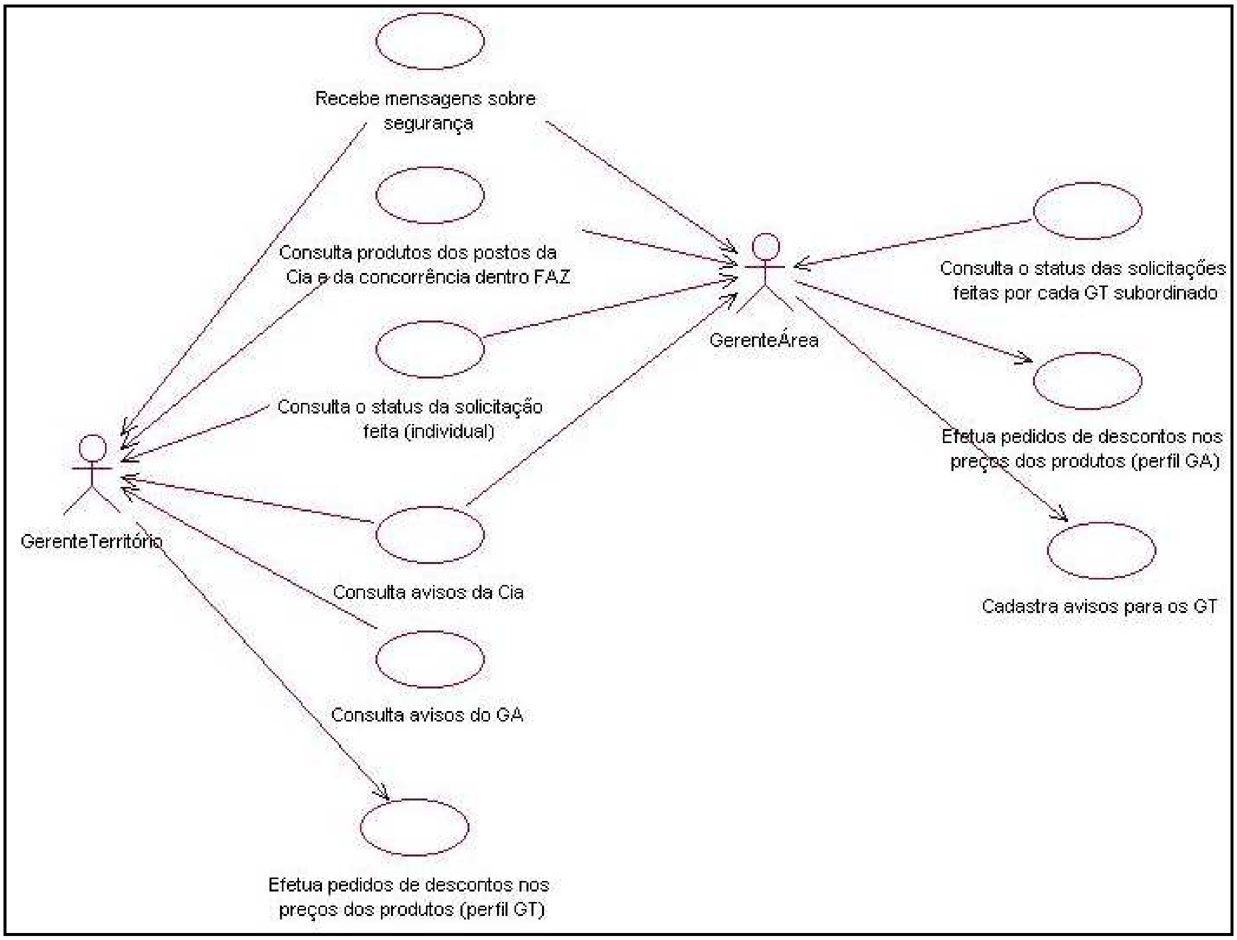


Figura 4-1 – Casos de Uso do sistema

Descrição dos casos de uso			
Nº do caso de uso	Iniciador do caso	Nome do caso de uso	Descrição do caso de uso
1	GT ou GA	Recebe mensagens sobre segurança	Mensagem inicial do sistema inclui o aviso sobre segurança
2	GT ou GA	Consulta produtos dos postos da cia e da concorrência dentro da FAZ	Com base no perfil do usuário (GT/GA), é permitida uma consulta aos produtos existentes em cada posto de gasolina existente na localidade de interesse.
3	GT ou GA	Consulta o status da solicitação feita (individual)	O usuário visualiza a situação do(s) seu(s) pedido(s) de desconto feito(s).
4	GT ou GA	Consulta avisos da cia	Mensagem inicial do sistema inclui além da mensagem sobre segurança, o aviso da cia informado pelo supervisor do usuário do sistema.
5	GT	Consulta avisos do GA	Mensagem inicial do sistema inclui também o aviso do supervisor do usuário do sistema.
6	GT	Efetua pedidos de descontos nos preços dos produtos (perfil GT)	O revendedor poderá solicitar que o preço de revenda do(s) produto(s) seja reduzido para um valor acordado com o revendedor. Esses pedidos estão restritos àqueles postos sob sua responsabilidade.
7	GA	Consulta o status das solicitações feitas por cada GT subordinado	A fim de possibilitar um acompanhamento gerencial da região, ao GA é permitida uma consulta abrangente de todos os pedidos feitos pelos seus subordinados.
8	GA	Efetua pedidos de descontos nos preços dos produtos (perfil GA)	O GA poderá efetuar pedidos de descontos em qualquer posto que esteja sob responsabilidade de seus subordinados, na sua área.
9	GA	Cadastra avisos para os GT	Possibilita que o GA cadastre mensagens para seus subordinados.

4.2.MODELO DE CLASSE

As principais classes obtidas através da análise foram:

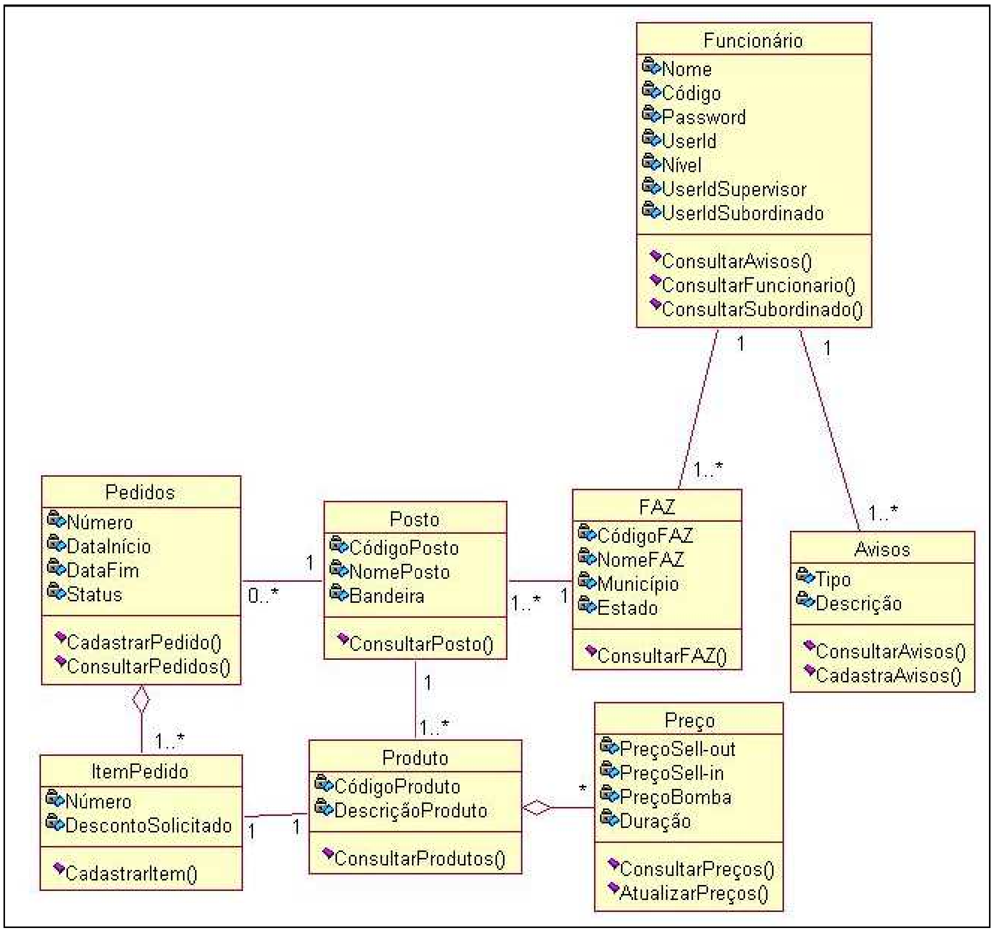


Figura 4-2 – Classes do FMMIS

Acima, temos as classes documentadas juntamente com seus relacionamentos. Não foi feito um questionamento conceitual profundo, evitando impor demasiada complexidade ao estudo.

A classe funcionário possui atributos que permitem caracterizar como uma pessoa é identificada dentro de uma organização, como nome, código (seria, por exemplo, a matrícula do funcionário), nível (identifica a posição hierárquica), userid e password (permitem que a pessoa tenha acesso aos recursos computacionais oferecidos) e useridsupervisor e useridsubordinado (identificam o supervisor e o subordinado, se existentes). Os métodos existentes *consultar avisos()* permite que o funcionário busque os avisos da cia e do supervisor para ele. Os métodos *consultar funcionário() e consultar subordinado()* servem de base para se conhecer melhor o funcionário em estudo.

A classe aviso engloba os avisos da cia e do supervisor para o usuário do sistema. Os avisos são identificados pelo tipo e uma descrição da mensagem em si. O usuário pode consultar avisos e/ou cadastrar avisos.

A classe de nome FAZ – filial, área e zona – caracteriza uma região geográfica que encontra-se sob responsabilidade do funcionário. A mesma possui um código, nome, e encontra-se em um dado estado e município. O método *consultarFAZ()* permite que sejam obtidos os atributos correspondentes a esta classe.

Uma FAZ engloba um ou mais postos. A classe posto é identificada por um código, nome e bandeira. Bandeira de um posto é a identificação da marca da empresa no mercado. O método *consultarPosto()* permite que sejam obtidos os atributos correspondentes a esta classe.

Esse sistema permite que sejam cadastrados solicitações de desconto na compra dos produtos (combustíveis) oferecidos pela companhia distribuidora do mesmo. Desta forma, a classe pedido está diretamente relacionada a classe posto, podendo ou não haver solicitações de descontos. Entretanto caso haja o pedido, o mesmo recebe uma identificação, status da solicitação (por exemplo, aprovado, recusado ou em análise) e dura por um período compreendido entre a data de início e fim. Fez-se necessário a inclusão dos métodos cadastrapedido() e consultapedido() para implementar as funcionalidades desta classe.

A classe itempedido permite que tenha-se a idéia de que cada pedido tem a ele associado um ou mais itens de pedido. Cada item de pedido possui um número identificador do mesmo e o valor do preço com o desconto. O método cadastraitem() implementa a solicitação de pedido de desconto.

A classe produto se relaciona tanto com o itempedido quanto com a classe posto. Em um posto tem-se um ou mais produtos presentes e um item de pedido possui um produto somente. O produto possui um código e uma descrição associados. O método *consultarProduto()* permite que sejam obtidos os atributos correspondentes a esta classe.

A classe preço possui os seguintes atributos: preço bomba, preço sell-in, preço sell-out e duração. Este último atributo indica se um determinado produto encontra-se em promoção ou não, e caso esteja indica por quanto tempo, em dias, aquele preço está em vigor. O preço bomba é aquele que o consumidor adquiri o produto. O preço sell-out é aquele preço pago pelo posto à companhia distribuidora de combustíveis. O preço sell-in é o preço que a companhia distribuidora paga à refinaria pelo produto. Os métodos correspondentes a esta classe são o *consultarPreço()* e *atualizarPreço()*.

4.3. Diagramas de Interação

Para entendermos as operações correspondentes às classes, é necessária a elaboração de diagramas de interação para cada caso de uso do sistema. As janelas e caixas de diálogo serão necessárias para fornecer uma interface com os atores.

Os diagramas de interação serão detalhados no apêndice 2.

Tendo o modelo de classes completo, incluindo atributos e operações, fez-se a implementação utilizando o banco de dados MySql e a linguagem PHP.

Neste capítulo sobre implementação do sistema serão descritos os scripts de criação das páginas em *PHP* e *WML* e a construção do banco de dados *FMMIS* com instruções de *MySQL.*

5.1.BANCO DE DADOS

Ao se implementar um sistema modelado de modo orientado a objetos em um banco de dado não orientado a objetos, algumas adaptações foram feitas.

5.1.1. CRIAÇÃO E CARGA INICIAIS DO BANCO DE DADOS

A criação do banco de dados FMMIS foi feita de forma automática, num *script* SQL (chamado "fmmis.sql"), que:

- Apaga o banco anterior de mesmo nome
- Cria as tabelas
- Adiciona índices e chaves nos campos correspondentes
- Adiciona os usuários especiais (admin e guest)
- Faz a carga inicial do banco a partir dos arquivos texto (extensões ".txt")

O arquivo “fmmis.sql” encontra-se na para referência, já que se trata de toda a estrutura do banco de dados.

```
###################################################
# Universidade Federal do Rio de Janeiro
# Departamento de Engenharia Eletrônica e Computação
# Projeto Final - [Fuels Marketing Mobile Information System]
#
# Professores Orientadores: Aloysio P. Pedrosa
#                                         Antonio  Cláudio Gómez de Souza
#
# Autor: Ingrid Teixeira
# Email: ingrid.teixeira@ufrj.br
# Data: 21/06/2002
# Descricao: Script SQL para criação do banco FMMIS
####################################################
# ==============================================
# Etapa 1: Criação do Banco fmmis no MySQL (substituindo eventuais antigos)
#===============================================
#
DROP DATABASE fmmis;
CREATE DATABASE fmmis;
# ===============================================
# Etapa 2: Criação de usuários de acesso/administração
# ===============================================
#
# [Usuário 1: fmmis_dba com privilégios totais (admin)]
#
GRANT ALL ON fmmis.* TO fmmis_dba@localhost IDENTIFIED BY "qwerty";
#
# [Usuário 2: fmmis_www com privilégios de uso (guest)]
#
GRANT FILE, SELECT, UPDATE ON FMMIS.* TO fmmis_www@localhost
IDENTIFIED BY "asdfgh";
#
# ===============================================
# Etapa 3: Criação das tabelas do banco de dados
# ===============================================
USE fmmis;
#
# [Tabela 1: t01_employee] // Com índices adicionados
# (Esta tabela contém as informações gerais do usuário)
#
CREATE TABLE t01_employee
        (
        EMPLOYEECODESMALLINT        NOT        NULL        PRIMARY        KEY
```

```
AUTO_INCREMENT,
        USERIDVARCHAR(7),
        NAMEVARCHAR(30),
        PASSWORDVARCHAR(16),
        LEVELTINYINT NOT NULL,
        S_USERIDVARCHAR(7)
        );
ALTER TABLE t01_employee ADD INDEX (S_USERID);
#
# [Tabela 2: t02_faz] // Com índices adicionados
# (Esta tabela contém as informações das áreas de vendas de casa funcionário)
#
CREATE TABLE t02_faz
        (
        FAZCODESMALLINT NOT NULL PRIMARY KEY,
        NAMEVARCHAR(30),
        DISTRICTVARCHAR(15),
        STATEVARCHAR(2),
        USERIDVARCHAR(7)
        );
ALTER TABLE t02_faz ADD INDEX (USERID);
#
# [Tabela 3: t03_product] // Com índices adicionados
# (Esta tabela contém as informações dos produtos disponíveis em cada Ponto de Venda)
#
CREATE TABLE t03_product
        (
       PRODUCTCODE     SMALLINT NOT NULL,
        NAMEVARCHAR(20),
        SELLOUTPRICEDECIMAL(1,3),
        PUMPPRICEDECIMAL(1,3),
        CATEGORYTINYINT NOT NULL,
        DURATIONTINYINT,
       GASSTATIONCODE  SMALLINT NOT NULL,
        ONSALEENUM('S','N') DEFAULT 'S'
        );
ALTER TABLE t03_product ADD INDEX (PRODUCTCODE,GASSTATIONCODE);
#
# [Tabela 4: t04_gstation] // Com índices adicionados
# (Esta tabela contém as informações dos Pontos de Venda)
#
CREATE TABLE t04_gstation
        (
        GASSTATIONCODESMALLINT NOT NULL PRIMARY KEY,
        NAMEVARCHAR(20),
        STATEVARCHAR(2),
        FLAGVARCHAR(15),
```

```
        FAZCODESMALLINT NOT NULL,
        DISTRICTVARCHAR(15)
        );
ALTER TABLE t04_gstation ADD INDEX (STATE, FAZCODE);
#
# [Tabela 6: t06_req]
# (Esta tabela contém as informações dos pedidos feitos de desconto)
#
CREATE TABLE t06_req
        (
        REQCODESMALLINT NOT NULL PRIMARY KEY,
        REQUSERIDVARCHAR(7),
        STATUSVARCHAR(15),
        DATEDATE,
       GASSTATIONCODE  SMALLINT NOT NULL UNIQUE,
        PRODUCTCODESMALLINT,
        DISCOUNTPRICEDECIMAL(1,3)
        );
#
# [Tabela 7: t07_message]
# (Esta tabela contém as mensagens do sistema)
#
CREATE TABLE t07_message
        (
        MESSAGECODETINYINT NOT NULL PRIMARY KEY,
        DESCRIPTIONVARCHAR(100),
        FROMUSERIDVARCHAR(7),
        TOUSERIDVARCHAR(7)
        );
#
# ==============================================
# Etapa 4: Adição de dados iniciais às tabelas
# ==============================================
#
INSERT INTO t01_employee VALUES (1,'iteixei','Ingrid Teixeira',password('teste'),21,'roamara');
INSERT INTO t01_employee VALUES (2,'rkcardo','Renata Cardoso',password('teste'),21,'roamara');
INSERT INTO t01_employee VALUES (3,'roamara','Ricardo Amarante',password('teste'),22,NULL);
INSERT INTO t01_employee VALUES (4,'mclemos','Miriam Lemos',password('teste'),21,'mbaccar');
INSERT INTO t01_employee VALUES (5,'mbaccar','Mauro Baccar',password('teste'),22,NULL);
#
INSERT INTO t02_faz VALUES (12345,'Botafogo','Rio de Janeiro','RJ','iteixei');
INSERT INTO t02_faz VALUES (12346,'Flamengo','Rio de Janeiro','RJ','iteixei');
```

```
INSERT INTO t02_faz VALUES (12347,'Meier','Rio de Janeiro','RJ','rkcardo');
INSERT INTO t02_faz VALUES (12348,'Madureira','Rio de Janeiro','RJ','mclemos');
#
INSERT INTO t03_product VALUES (1,'Gasolina Comum',1.021,1.031,1,10,1,'S');
INSERT INTO t03_product VALUES (2,'Gasolina Maxxi',1.121,1.131,1,3,1,'S');
INSERT INTO t03_product VALUES (3,'Alcool',0.921,0.931,2,NULL,2,'N');
INSERT INTO t03_product VALUES (4,'Diesel',1.921,1.931,3,NULL,3,'N');
INSERT INTO t03_product VALUES (1,'Gasolina Comum',1.321,1.331,1,10,3,'S');
INSERT INTO t03_product VALUES (2,'Gasolina Maxxi',1.421,1.431,1,3,4,'S');
INSERT INTO t03_product VALUES (3,'Alcool',1.021,1.031,2,NULL,4,'N');
INSERT INTO t03_product VALUES (4,'Diesel',1.911,1.921,3,NULL,4,'N');
#
INSERT INTO t04_gstation VALUES (1,'Manequinho','RJ','Esso',12345,'Botafogo');
INSERT INTO t04_gstation VALUES (2,'Espanhol','RJ','BR',12346,'Flamengo');
INSERT INTO t04_gstation VALUES (3,'Rosas','RJ','Texaco',12347,'Meier');
INSERT INTO t04_gstation VALUES (4,'Redentor','RJ','Ipiranga',12348,'Madureira');
#
INSERT INTO t07_message VALUES (1,'Bem-vindo',NULL,NULL);
INSERT INTO t07_message VALUES (2,'Mensagem Supervisor','roamara','iteixei');
INSERT INTO t07_message VALUES (3,'Passe no RH','mbacar','mclemos');
INSERT INTO t07_message VALUES (4,'Recursos Humanos','roamara','rkcardo');
```

5.1.2. TABELAS E SEUS RELACIONAMENTOS

As tabelas estabelecidas foram:

- T01_Employee : Tabela base dos Usuários, contém informações capazes de identificar um funcionário. Relaciona-se com as tabelas de T02_FAZ e T07_Message.

- T02_FAZ : Tabela que contém as informações relativas às Filiais, Área e Zona sob a responsabilidade do empregado. Relaciona-se com as tabelas T01_Employee e T04_GStation.

- T07_Message : Tabela que contém as mensagens cadastrados pelos Gerente de Área aos Gerente de Território, mensagem inicial da companhia e mensagens sobre segurança. Relaciona-se com a tabela T01_Employee.

- T04_GStation : Tabela que armazena os registros dos postos de gasolina. Relaciona-se com a tabela T02_FAZ, T06_Req e T03_Product.

- Tabela T06_Req : Tabela que contém os registros relativos aos pedidos relacionados a cada posto de gasolina. Devido a baixa complexidade exigida pelo sistema não construiu-se a tabela de itens de pedido. Relaciona-se com as tabelas T04_GStation e T03_Product

- T03_Product : Tabela que armazena as informações relativas aos produtos comercializados em cada posto. Relaciona-se com a tabela T04_GStation.

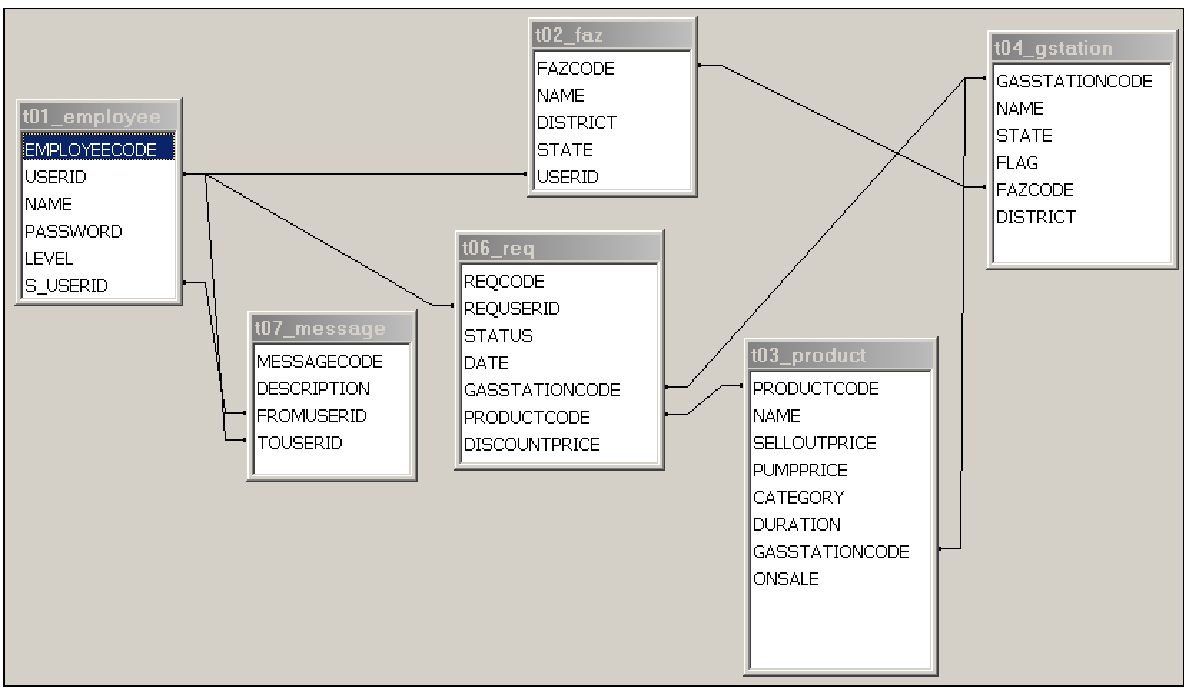


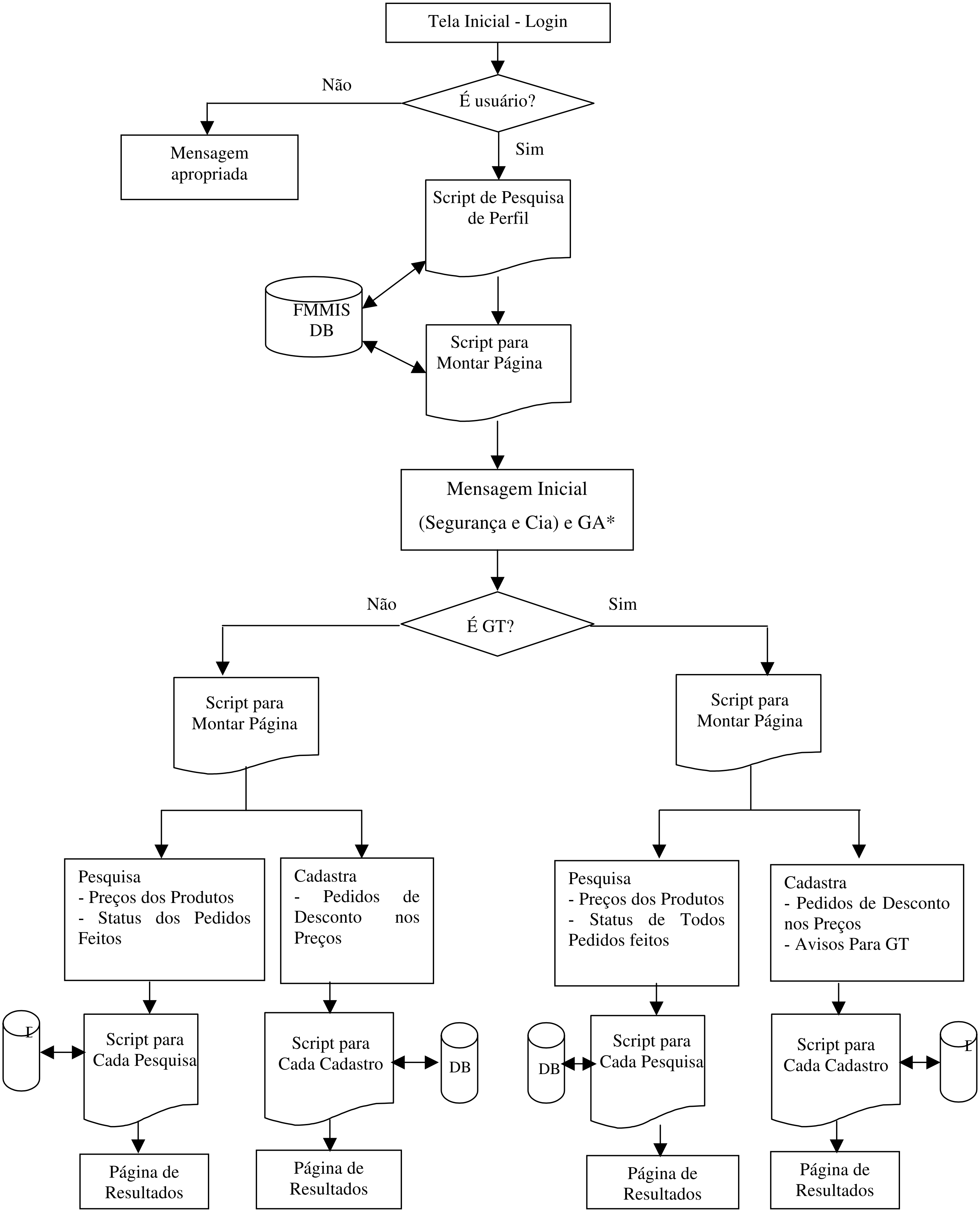
Figura 5-1– Tabela com os relacionamentos do Banco de Dados do FMMIS

5.2. PHP

O cliente PHP é o grande responsável por toda a interface com o usuário do FMMIS, por meio de páginas WML geradas dinamicamente, portanto, acessíveis via *browser* do celular.

As possibilidades de emprego do FMMIS utilizando o PHP são inúmeras, no limite do tempo disponível e da imaginação do programador, pelo fato de o PHP ser uma linguagem profundamente poderosa e cheia de recursos adicionais que podem ser incorporados a ela.

As telas foram escritas em WML e PHP obedecendo o fluxograma a seguir. O fluxograma foi construído de maneira didática, para que facilitasse a compreensão da aplicação. As telas são as caixas retangulares, os scripts estão indicados através do símbolo de documento e o banco de dados do aplicativo é também indicado neste fluxograma.


Tela Inicial - Login
Não
É usuário?
Sim
Mensagem apropriada
Script de Pesquisa de Perfil
FMMIS DB
Script para Montar Página
Mensagem Inicial
(Segurança e Cia) e GA*
Não
É GT?
Sim
Script para Montar Página
Script para Montar Página
Pesquisa
- Preços dos Produtos
- Status dos Pedidos Feitos
Cadastra
- Pedidos de Desconto nos Preços
Pesquisa
- Preços dos Produtos
- Status de Todos Pedidos feitos
Cadastra
- Pedidos de Desconto nos Preços
- Avisos Para GT
Script para Cada Pesquisa
Script para Cada Cadastro
DB
DB
Script para Cada Pesquisa
Script para Cada Cadastro
Página de Resultados
Página de Resultados
Página de Resultados
Página de Resultados

5.2.1. Telas do Sistema

As telas a seguir foram obtidas através do simulador *yourWap* obtido gratuitamente na *Wireless Companion*. O aparelho utilizado foi da empresa *Siemens*.

5.2.2. Tela Inicial do Sistema

A tela inicial do sistema irá obter as informações sobre a userid e senha do usuário e verificar se as mesmas são válidas.

Figura 5-2 – Telas Iniciais FMMIS

Abaixo encontra-se o código que gerou as páginas da figura 5-2 acima.

```
<?php
#########################################################          #
#       Universidade do Brasil - UFRJ                              #
#Escola Politecnica - Engenharia Eletrônica & da Computação        #
#                                                                  #
#Aluna: Ingrid Teixeira                                            #
#E-mail: ingrid.teixeira@ufrj.br                                   #
#Profs.Orientadores: Aloysio C. P. Pedroza                         #
#                          Antonio Claudio                         #
#                                                                  #
#Nome Arquivo:index.wml                                            #
#Funcionalidade:                                                   #
#       Tela Inicial do Sistema.                                   #
#       Elimina o registro de variáveis : user                     #
#Data Ultima Modificacao: 28 de março de 2002                      #
#########################################################          #

        include("config.php");
        session_start();
        if (session_is_registered ('user')) session_unregister ('user');
        top("Welcome to FMMIS");
```

```
?>
	<p align="center">
	Fuels Marketing
	</p>
	<p align="left">
	 Userid: <input name="user" type="text" />
	 Password: <input name="pass" type="password" />
	<do type="accept" label="Send">
	 <go href="perfil.wml" method="post">
	 <postfield name="user" value="$user" />
	 <postfield name="pass" value="$pass" />
	 </go>
	 </do>
	</p>
<?php
   bot();
?>
```

O arquivo *config.php* contém alguns parâmetros gerais de conexão com o banco de dados e funções que implementam código que se repetem em todas as páginas WML.

```
<?php
####################################################################### #
#	Universidade do Brasil - UFRJ                                          #
#Escola Politecnica - Engenharia Eletrônica & da Computação               #
#                                                                         #
#Aluna: Ingrid Teixeira                                                   #
#Profs.Orientadores: Aloysio C. P. Pedroza                                #
#                    Antonio Claudio                                      #
#                                                                         #
#Nome Arquivo:config.php                                                  #
#Funcionalidade:                                                          #
#	Configurações para conexão com o banco de dados FMMIS                  #
#                                                                         #
#Data Ultima Modificacao: 28 de março de 2002                             #
####################################################################### #

####################################################################### #
#                                                                         #
# Cabeçalho das funções utilizadas:                                       #
#                                                                         #
# void mysql_close(MYSQL *mysql)                                          #
```

```
# MYSQL *mysql_connect(MYSQL *mysql, const char *host,                #
#       const char *user,const char *passwd)                          #
#                                                                     #
################################################################### #

function conecta_banco()
{
        $filename = "config.ini";
        $filepath = "C:\\Apache\\htdocs\\" . $filename;
        $fd       = fopen($filepath, "r");
        $db_name  = "fmmis";
        while (!feof($fd))
        {
        $temp    = fgets($fd,4096);
        $host    = fgets($fd,4096);
        $temp    = fgets($fd,4096);
        $db_user = fgets($fd,4096);
        $temp    = fgets($fd,4096);
        $db_pass = fgets($fd,4096);
        }
        echo $temp . "<br />";
        # Conexao ao banco local! Devera mudar o endereco IP para o endereco
onde ficara hospedado o banco de dados.
        $connection=mysql_connect("www2.del.ufrj.br","fmmis_www","asdfgh")
or die ("Não foi possível conectar a base de dados");
        return($connection);
}
function seleciona_db($connection)
{
        # Conexao ao banco de dados Mysql FMMIS
        $db=mysql_select_db("fmmis",$connection) or die ("Não foi possível
conectar ao banco de dados");
        return($db);
}
function top($title)
{
        echo "<!DOCTYPE wml PUBLIC \"-//WAPFORUM//DTD WML
1.1//EN\" \"http://www.wapforum.org/DTD/wml_1.1.xml\">";
        echo "<wml>";
        echo "<card id=\"main\" title=\"$title\">";
}
function bot()
{
        echo "</card>";
        echo "</wml>";
}
?>
```

5.2.3. TELA DE ABERTURA

Nesta tela, o usuário receberá as informações cadastrados no banco de dados FMMIS da companhia, sobre segurança e do supervisor, se existente. A userid do supervisor aparece em negrito, logo após as duas mensagens iniciais.

Figura 5-3 – Tela de Abertura do FMMIS

No aparelho escolhido a opção de consulta aparece no canto inferior direito, com o nome de *cons* e a opção de cadastro é visualizada se escolhe-se a opção *menu* e em seguida a opção *cada.*

5.2.4. TELA DE CONSULTAS

A partir desta tela o usuário poderá optar pela consulta aos preços dos produtos ou status dos pedidos de descontos.

Figura 5-4 – Tela de Consultas do FMMIS

De acordo com o perfil do usuário e atende aos requisitos de especificação do sistema, as telas subseqüentes à escolha feita será dinamicamente construída para o mesmo.

5.2.5. TELA DE ESCOLHA DA FAZ

São exibidas a(s) FAZ sob responsabilidade do usuário do sistema. O usuário é levado de modo intuitivo a selecionar uma FAZ de interesse. No caso do perfil do usuário ser GA, são exibidos todas as FAZ sob responsabilidade de cada um de seus subordinados.

Figura 5-5 – Tela de escolha da FAZ

5.2.6. TELA DE ESCOLHA DO POSTO

A seguir serão exibidas a listagem de todos os postos cadastrados dentro da FAZ do usuário. No caso do perfil do usuário ser GA, são exibidos todos os postos sob responsabilidade de cada um de seus subordinados.

Figura 5-6 – Tela de escolha do Posto

5.2.7. TELA COM O RESULTADO FINAL DA CONSULTA POR PREÇOS DE PRODUTOS E STATUS DAS SOLICITAÇÕES

A tela final obtida como resultado da consulta dos preços encontra-se a seguir.

Figura 5-7 – Tela final da Consulta por preços dos produtos

A tela final obtida como resultado do status das solicitações encontra-se a seguir.

Figura 5-8 – Tela final da Consulta do status das solicitações feitas

5.2.8. TELA DE CADASTROS

No perfil GT, a única opção disponível é a de cadastro de pedidos de descontos nos produtos existentes nos postos de gasolina. Como o usuário deve efetuar a sua solicitação a um determinado produto, serão exibidas as mesmas telas mostradas anteriormente para a escolha da FAZ e Posto.

Figura 5-9 – Tela de Cadastros de Pedidos de descontos, perfil GT

A seguir é exibida a tela de escolha do produto existente no posto de interesse.

Figura 5-10 – Tela de escolha do produto

No capítulo anterior, 5, mostrou-se as telas do sistema de modo estático. Neste capítulo serão descritas as simulações dos casos de uso mais freqüentes durante a utilização do sistema.

Após a identificação feita na tela de abertura do sistema, o usuário possui dois caminhos a seguir: consultas ou cadastros gerais. Abaixo serão descritos três cenários de uso do sistema, aqueles com mais interações com os usuários.

6.1. CADASTRO DE PEDIDOS DE DESCONTOS

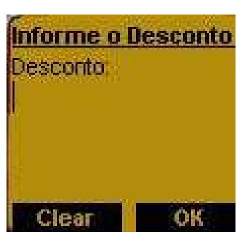
Na simulação realizada, o usuário possui o perfil Gerente de Território. Portanto, só lhe é mostrada a opção de cadastro de pedidos de descontos dos produtos.

Ao escolher a opção *link* será exibida a tela que possibilitará a escolha dos bairros existentes dentro da sua área de atuação (Filial,Área e Zona) de vendas.

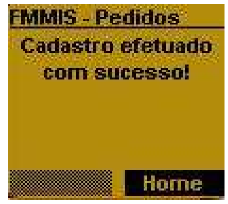
Novamente ao escolher a opção *link* será exibida a tela que possibilitará a escolha dos postos de gasolina cadastrados naquele bairro. A lista de postos exibirá os postos da companhia do funcionário existentes na FAZ escolhida. Caso não haja postos da cia cadastrados no sistema, uma mensagem apropriada surgirá.

Nesse momento, o usuário fará a escolha do produto cadastrado para o posto em questão e para o qual deseja fazer uma solicitação de desconto no preço de compra do produto. O usuário informa qual o preço de compra desejado para o produto.

A seguir será exibida uma tela para que o usuário possa informar o preço do produto:



É exibida, então, uma mensagem final ao usuário.



6.2. CONSULTA DOS PEDIDOS FEITOS – PERFIL GT

O sistema permite duas consultas ao usuário com o perfil GT, a de preços dos produtos existentes na sua área de atuação e a dos status dos pedidos realizados. No caso da consulta dos status dos pedidos realizados, o usuário deverá optar pelo link correspondente na tela exibida abaixo:

A seguir serão novamente exibidas as telas mostradas anteriormente para a escolha da FAZ de interesse e do posto.

A seguir será exibida uma listagem de todos os pedidos cadastrado para aquele posto, se existirem solicitações de desconto é claro.

6.3. Cadastro de mensagens

Se o usuário do sistema for perfil GA, ou seja, gerente de território, ele possui alguns GTs sob sua responsabilidade. Deste modo a ele é permitido realizar cadastro de avisos para seus subordinados. É exibida a seguinte tela:

Em seguida, o usuário deverá informar a userid do GT para o qual deseja enviar a mensagem. O sistema irá posteriormente criticar a valida da userid informada, verificando se a mesma está correta e se esse GT realmente é subordinado ao usuário com o perfil GA que está utilizando o sistema.

Após selecionarmos *ok*, exibe-se a tela onde podemos informar a mensagem em si.

O aplicativo exibe ao usuário a mensagem cadastrada, a fim de validar a mesma.

Estando a mensagem correta, é exibida a mensagem final de cadastro realizado com sucesso.

O FMMIS (Sistema de Informações Móveis para o Departamento de Marketing de Combustíveis) mostrou-se de um projeto que vem de encontro a necessidade de uma empresa dinâmica que deseja manter seus clientes sempre bem atendidos. É um embrião para futuras tarefas que poderão, sem problemas, ser agregadas a ele em qualquer um de seus níveis.

A abrangência do projeto, mesmo em se considerando o nível de desenvolvimento da primeira versão, é grande e significativa , permitindo diversas funcionalidades e aplicações.

Embora todo o projeto tenha sido desenvolvido com a clara finalidade de tornar-se o mais fechado e funcional possível – objetivo que, dentro das expectativas, atingiu, sem dúvida – logicamente algumas funcionalidades pensadas escapavam ao escopo dessa versão inicial, e devem ser listadas e comentadas para consulta e implementação posteriores.

7.1. CONTINUAÇÕES E MELHORIAS A SEREM AGREGADAS AO FMMIS

A seguir são exibidas algumas idéias para que o projeto possa sofrer melhorias futuras.

7.1.1. CRIPTOGRAFIA PARA A TROCA DE MENSAGENS COM O SERVIDOR

Hoje em dia, e cada vez mais, a segurança da informação se faz necessária. Dessa forma, poderia ser implementada uma biblioteca para dispositivos móveis que provesse tal funcionalidade no ambiente WAP.

Converse, T.; Park, J. *PHP 4 – A Bíblia.* Editora Campus.

Floyd, R.; Houssel, B.; Tatit, C. (1998) *Mobile web access using eNetwork We Express,* IEEE Personal Comunications, (5).

Housel, B.; Lindqist, D. (1996) *WebExpress: A system for optimizing web browsing in a wireless environment.* Proc. ACM/IEEE MobiCom'96 conference, Rye, NY, EUA.

[HTML4] "HTML 4.0 Specification, W3C Recommendation 18-December-1997, REC-HTML40-971218", D. Raggett, et al., September 17, 1997. URL: http://www.w3.org/TR/REC-html40

King, P.; Hyland, T. (1997) *Handled device markup language specification. Unwired Planet.*

Liljeberg, M.; Alanko, T., Kojo, M.; Laamanen, H.; Raatikainen, K. (1995) *Optimizing world wide web for wekly connected mobile workstations: Na indirect approach,* proc. 2º. International Workshop on Services in Distributed and Network Environments, SDNE'95, Whistler, B.C., Canadá.

Liljeberg, M.; Helin, H.; Kojo, M.; Raatikainen, K.: *Mowgli WWW* (1996) *Improved usability of www in mobile WAN environments*, proc. IEEE Global Internet 1996, Londres, Inglaterra.

Maslakowski, M. *Aprenda em 21 dias MySQL.* Editora Campus.

[PushArchOverview] "WAP Push Architectural Overview", WAP Forum.

[PushOTA] "WAP Push OTA Protocol", WAP Forum.

Raggett, D.; LeHors, A.; Jacobs, I. (1998) *HTML 4.0 specifiaction,* W3C recommendation, REC-xml-19980424

[RFC2045] "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", N. Freed, et al., Novembro 1996. URL: http://www.rfc-editor.org/rfc/rfc2045.txt

[RFC2048] "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", N. Freed, et al., Novembro 1996. URL: http://www.rfc-editor.org/rfc/rfc 2048.txt

[RFC2396] “Uniform Resource Identifiers (URI): Generic Syntax”, T. Berners-Lee, R. Fielding, L. Masinter. Agosto 1998. URL: http://www.rfc-editor.org/rfc/rfc2396.txt

[RFC2616] "Hypertext Transfer Protocol - HTTP/1.1", R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999. URL: http://www.rfc-editor.org/rfc/rfc2616.txt

Schiller, J. *Mobile Communications*, 2000, Addison Wesley ed.

[STD0007] “Transmission Control Protocol”, J. Postel, Setembro 1981. URL: http://www.rfc-editor.org/rfc/std/std7.txt

[STD0013] “Domain Name System”, P. Mockapetris, November 1987.
URL: http://www.rfc-editor.org/rfc/std/std13.txt

[WAE] "Wireless Application Environment Specification", Version 2, WAP-236-WAESpec, WAP Forum.

[WAPCert] “WAP Certificate and CRL Profiles”, WAP Forum.

WAP Forum(1998a) *Wireless application protocol architecture specification,* WAP Forum, HTTP://www.wapforum.org/

[WDP] "Wireless Datagram Protocol Specification", WAP Forum.

[WIM] “WAP Identity Module Specification”, WAP Forum.

WML – Wireless Markup Language specification, WAP Forum, http://www.wapforum.org/

[W3C] World Wide Web Consortium (1998) *Extensible Markup Language (XML) 1.0 Specification,* W3C recommendation, REC-xml-19980210

[WPKI] "WAP Public Key Infrastructure Definition", WAP Forum.

Em todo o documento foram utilizados diversos acrônimos que podem, de maneira de fácil acesso, serem consultados a seguir:

ASP: *Active Server Pages*

CGI: *Common Graphics Interface*

CSS: *Client Side Script*

DEL: *Departamento de Eletrônica e Computação*

FAZ: *Filial, Área, Zona*

GA: *Gerente de Área*

GT: *Gerente de Território*

HTML: *Hypertext Markup Language*

HTTP: *Hypertext Transmission Protocol*

IP: *Internet Protocol*

PDU: *Protocol Data Unit*

PHP: *Pre Hypertext Processor*

SGBD: *Sistema Gerenciador de Banco de Dados*

SQL: *Structured Query Language*

SSS: *Server Side Script*

TCP: *Transmission Control Protocol*

UFRJ: *Universidade Federal do Rio de Janeiro*

WAP: *Wireless Application Protocol*

WWW: *World Wide Web*

WML: *Wireless Markup Language*

1. INTRODUÇÃO

1.1.FINALIDADE

Detalhamento das funcionalidades do software *FMMIS* a ser analisada pelo revisor técnico formal.

1.2.ESCOPO

FMMIS – Fuels Marketing Mobile Information System tem por objetivo agilizar o processo de atendimento ao revendedor de combustíveis, fornecendo aos telefones celulares dos funcionários remotos informações atualizadas e precisas sobre os preços dos produtos dos postos da companhia e da concorrência. Além disso, o sistema também possibilitará o acompanhamento da situação das requisições de desconto nos produtos feitas.

1.3.DEFINIÇÕES, ACRONISMOS E ABREVIATURAS

FMMIS – Fuels Marketing Mobile Information System, é o nome do sistema.
GT – gerente de território, é o funcionário da cia que tem por principal função supervisionar uma ou mais determinadas áreas. É ele quem solicita aos analistas de preços da Matriz os pedidos de descontos nos preços de compra dos combustíveis.
GA – gerente de área, é o funcionário
FAZ – filial, área e zona, é a região sob a responsabilidade do GT. Essa região é delimitada segundo razões geográficas e de mercado.

1.4. Referências

Os documentos referenciados na ERS serão: o plano de projeto para software, manuais sobre PHP, WMLScript, WML e MySQL.

1.5. Resumo

No item 3.1 que se segue, há a modelagem do sistema através do conceito de OO (Orientação a Objeto). São definidas classes, relacionamentos e heranças sobre as definições de algumas classes básicas para o funcionamento do *software* (posteriormente relatadas). Além disso, foram especificados métodos e membros pertinentes a cada classe, utilizados pelos casos de uso do sistema, em interfaces com o usuário final.

2. *Descrição Geral*

2.1. *Perspectiva do Produto*

O software *FMMIS* utilizará o *MySQL* como ferramenta de Gerenciamento de Banco de Dados. O MySQL além de armazenar os dados de interesse e disponibilizá-los para o sistema, irá também controlar os acessos a esses dados. Será estabelecida uma conexão com o servidor toda vez que o celular 3G do cliente acessar a página do sistema através do navegador. Desta forma, a correta instalação e funcionamento do MySQL no servidor são imprescindíveis.

2.2. *Funções do Produto*

O *FMMIS* define dois atores principais, o gerente de território(GT) e o gerente de área(GA). Suas funções são descritas a seguir.

O *Gerente de Território(GT)* fará os pedidos de descontos nos preços dos produtos, de acordo com o seu perfil. Também é possível ao GT consultar os preços praticados pelos postos de gasolina dentro da zona de influência, consultar o status da solicitação feita por ele e avisos do GA. Ele receberá mensagens sobre Segurança e avisos da Cia.

O *Gerente de Área(GA)* fará os pedidos de descontos nos preços dos produtos, de acordo com o seu perfil e fará o cadastro de avisos para os GT. Além das consultas aos preços praticados pelos postos da Cia e da concorrência dentro de sua FAZ, o GA pode consultar o status das solicitações feitas por ele próprio e pelos GT subordinados. Ele também receberá mensagens sobre Segurança e avisos da Cia.

2.3. Características do Usuário

O Gerente de Território é um usuário intermitente, de idade variável, de conhecimento mediano. O Gerente de Área é um usuário também ocasional, de conhecimento elevado com conhecimento de informática.

2.4. Restrições

O cliente utilizará um aparelho de telefone celular com hardware com pouca capacidade de processamento computacional. Esse fato restringe o projeto, impondo que o maior processamento da informação ocorra no servidor.

Outro fator a ser levado em consideração é o tamanho do *display* do aparelho do cliente. Deverá ser implementada uma interface que seja amigável com o usuário.

2.5. Pressupostos e Dependências

Disponibilidade de acesso ao servidor para a efetivação dos pedidos feitos pelos clientes.

3. *REQUISITOS ESPECÍFICOS*

3.1.REQUISITOS FUNCIONAIS

Foram necessárias 10 classes, com os atributos e métodos relacionados na tabela abaixo.

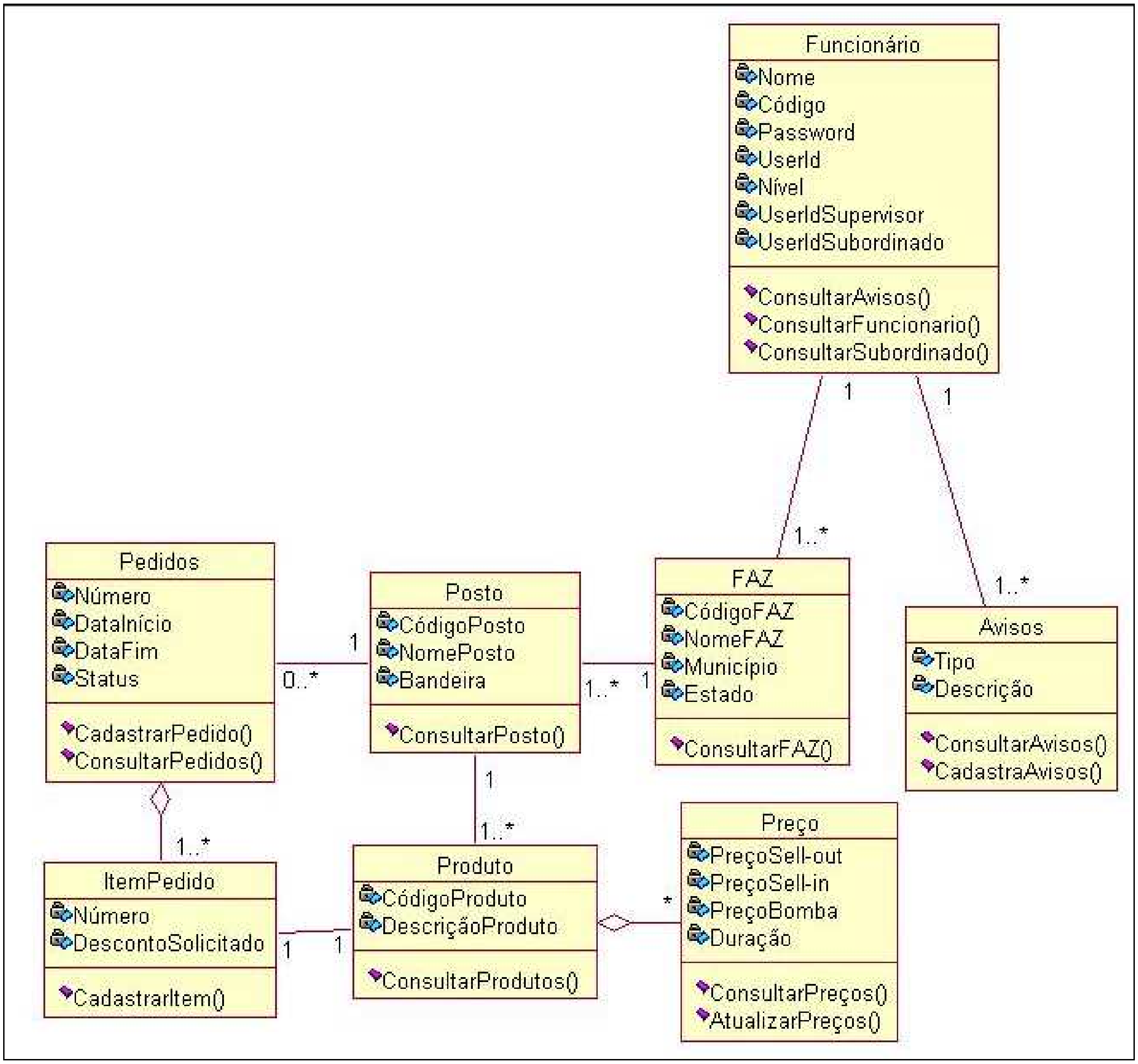
#	Nome	Atributos[6]	Métodos[7]
1	Funcionário	Nome	ConsultarAvisos()
		Código	ConsultarFuncionário()
		Password	ConsultarSubordinado()
		UserId	
		Nível	
		UserIdSupervisor	
		UserIdSubordinado	
2	FAZ	CódigoFAZ	ConsultarFAZ()
		NomeFAZ	
		Município	
		Estado	
3	Avisos	Tipo	ConsultarAvisos()
		Descrição	CadastrarAvisos()
4	Postos	CódigoPosto	ConsultarPosto()
		NomePosto	
		Bandeira	

[6] Todos os atributos são do tipo *private*.

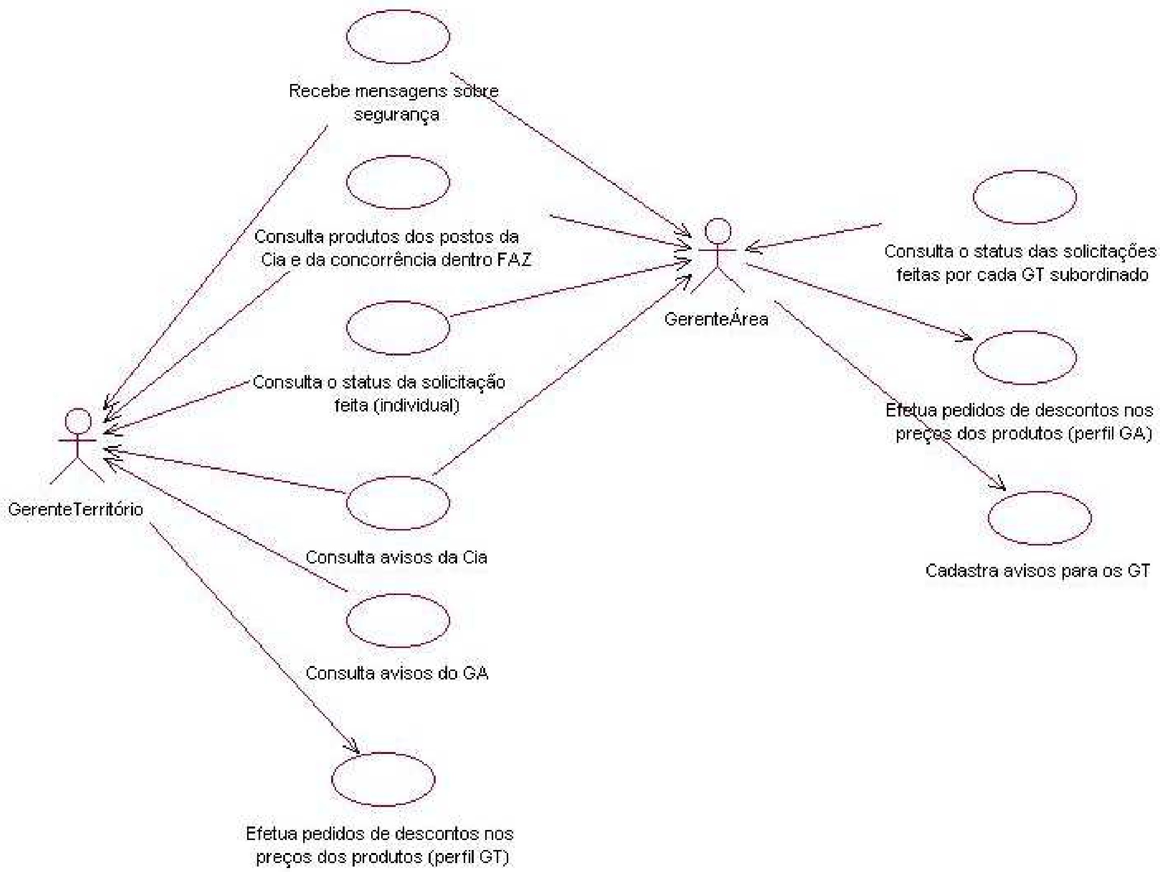
[7] Todos os métodos são do tipo *public*. Estão descritos nesta tabela somente os métodos de cada classe, não sendo considerados os métodos herdados de outras classes, caso haja herança.

#	Nome	Atributos[6]	Métodos[7]
5	Pedidos	Número	ConsultarPedidos()
		Data	CadastrarPedidos()
		Status	
6	ItemPedido	Número	CadastrarItem()
		Preço	
		DescontoSolicitado	
7	Produto	CódigoProduto	ConsultarProdutos()
		DescriçãoProduto	
8	Preço	Preço Sell-out	ConsultarPreços()
		Preço Sell-in	CadastrarPreços()
		Margem	

A seguir temos o Diagrama de Classes do sistema, com a definição dos relacionamentos entre as classes.

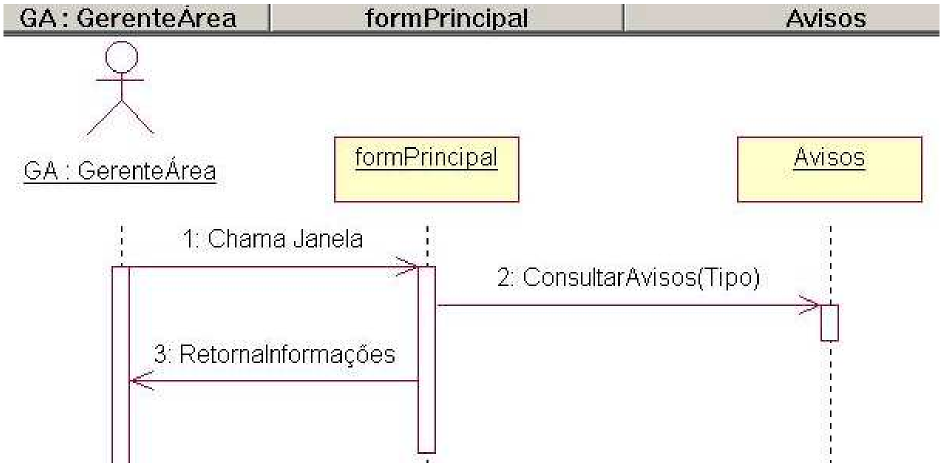


Serão apresentados os Casos de Uso e os atores para o sistema. O sistema terá dois atores, a saber: Gerente de Território e Gerente de Área, com os casos de uso abaixo mostrados.

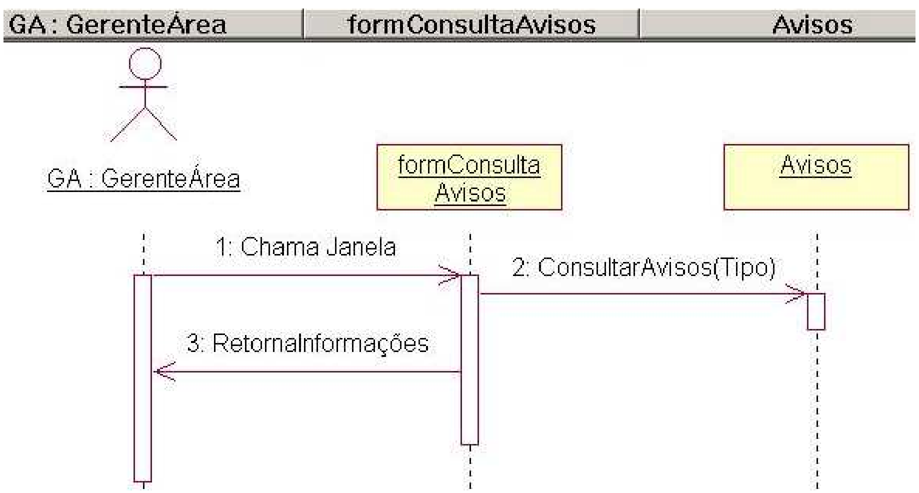


3.2. CASOS DE USO:

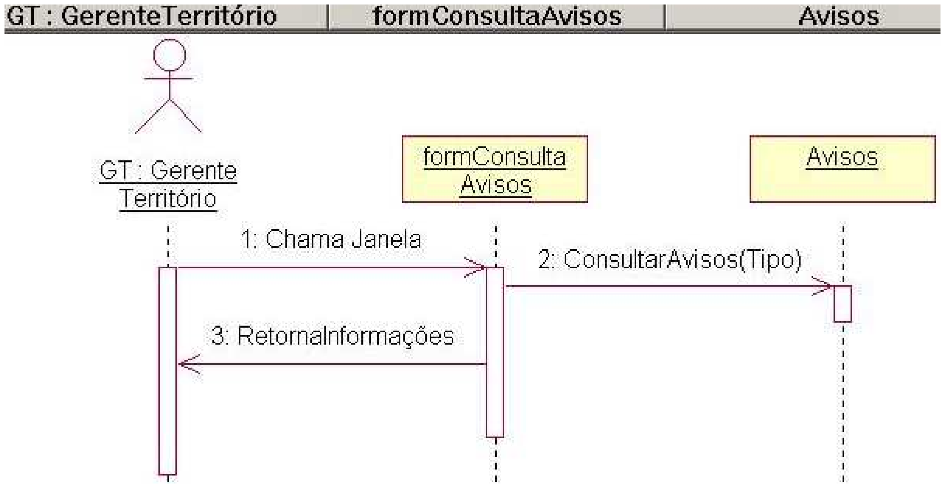
A) Recebe mensagem sobre segurança



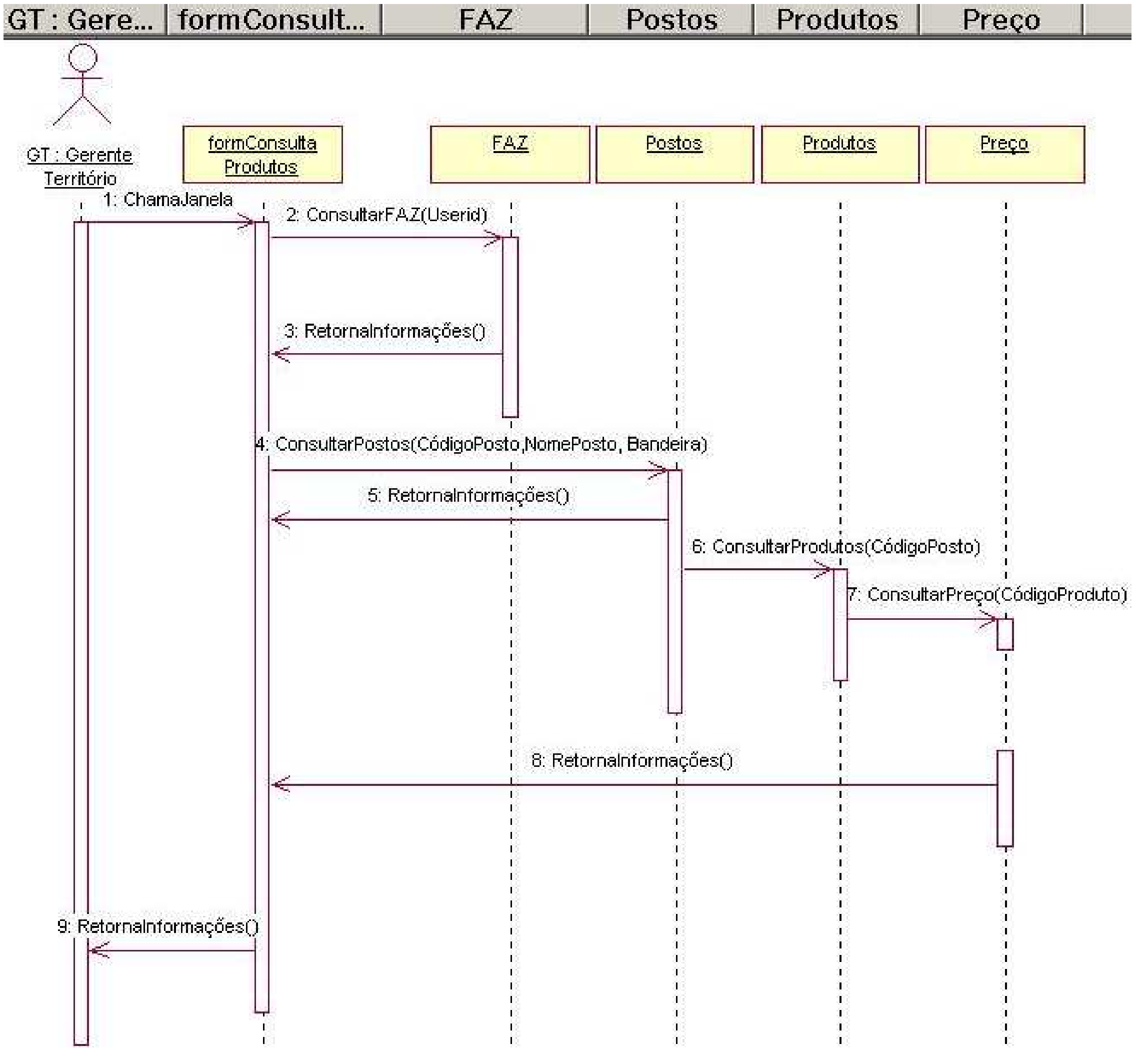
B) Consulta avisos da cia , perfil GA



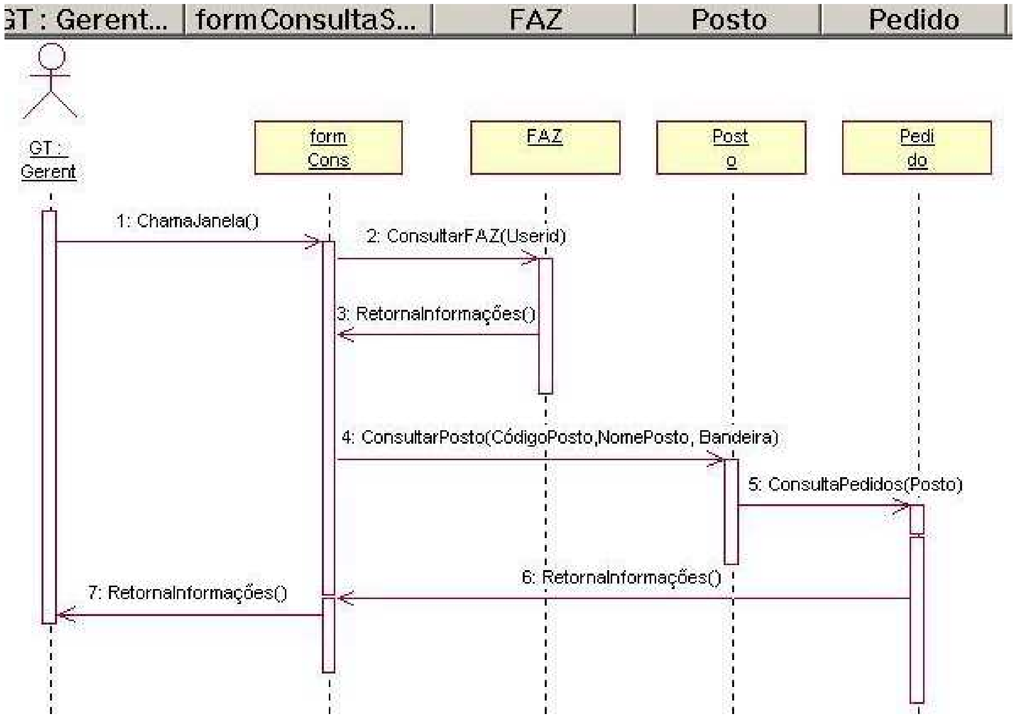
C) Consulta avisos da cia , perfil GT


GT : GerenteTerritório
formConsultaAvisos
Avisos
GT : Gerente Território
formConsulta Avisos
Avisos
1: Chama Janela
2: ConsultarAvisos(Tipo)
3: RetornaInformações

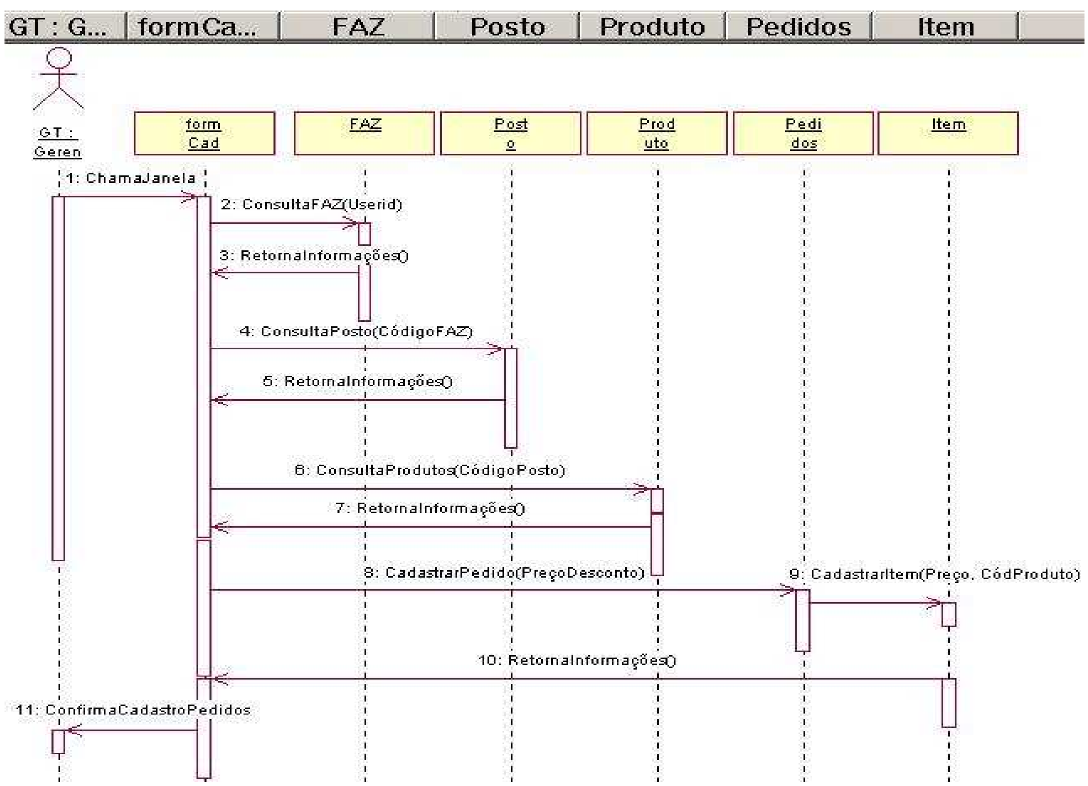
D) Consulta produtos dos postos



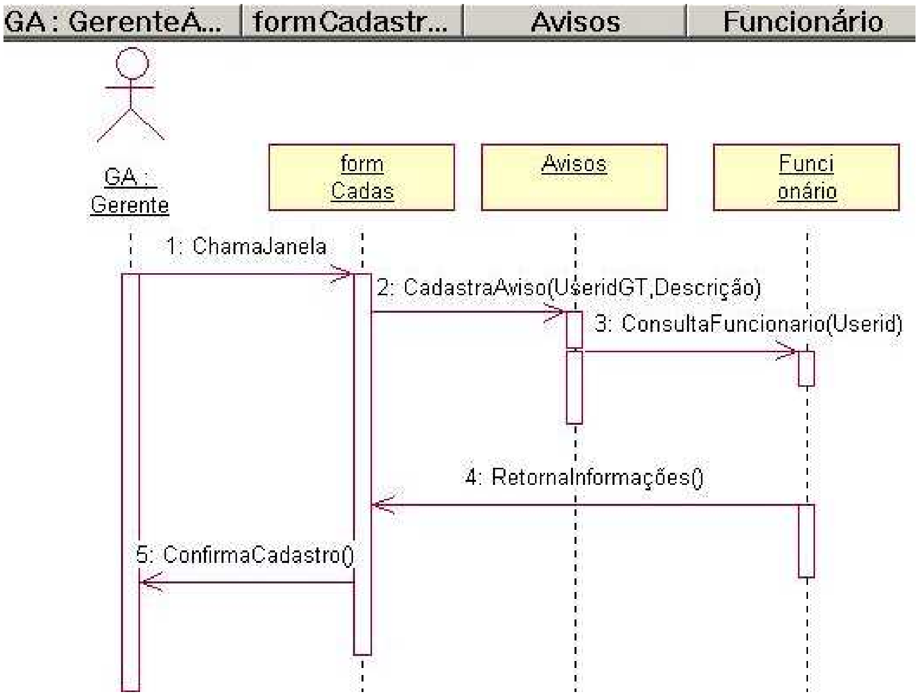
E) Consulta status das solicitações



F) Efetua pedidos de descontos dos produtos



G) Cadastra avisos para GT



3.3.INTERFACES EXTERNAS

3.3.1. INTERFACES DOS USUÁRIOS

a. Gerente de Área

a.1. Tela de Cadastro de Avisos para os GTs subordinados

Campos de Entrada de Dados: Gerente de Território e mensagem

Campos de Saída de Dados: n/a

Botões: Ok, sair

a.2. Tela de Cadastro de Pedidos de descontos nos preços dos produtos (perfil GA)

Campos de Entrada de Dados: Valor do Desconto

Campos de Saída de Dados: n/a.

Botões: OK, sair

a.3. *Tela de Consulta o status das solicitações feitas por cada GT*

Campos de Entrada de Dados: UserIdGT, FAZ, Posto

Campos de Saída de Dados: Posto, Status, Nome do Produto, Desconto

Botões: Return

b. Gerente de Território

b.1. Tela de Consulta o status das solicitações feitas

Campos de Entrada de Dados: FAZ, Posto

Campos de Saída de Dados: Posto, Status, Nome do Produto, Desconto

Botões: Return

b.2. Tela de Cadastro de Pedidos de descontos nos preços dos produtos

Campos de Entrada de Dados: Código do Produto, Valor do Desconto

Campos de Saída de Dados: Número do Pedido, Status

Botões: OK, Sair

c. Gerente de Território e Gerente de Área

c.1. Tela Principal

Campos de Entrada de Dados: UserId, Senha.

Campos de Saída de Dados: n/a

Botões: OK, Sair

c.2. Tela de Consulta de Mensagens

Campos de Entrada de Dados: Tipo (Cia, GA)

Campos de Saída de Dados : Mensagem

Botões: OK

c.3. Tela de Consulta aos Produtos dos Postos da Cia e da Concorrência dentro da FAZ

Campos de Entrada de Dados: Código Produto, FAZ

Campos de Saída de Dados: Preço, Duração, Código Produto, Bandeira Posto

Botões: OK, Sair

3.3.2. INTERFACES DE HARDWARE

Não aplicável.

3.3.3. INTERFACES DE SOFTWARE

Serão utilizados os protocolos de comunicação disponíveis tanto nos aparelhos celulares e nas centrais telefônicas.

3.3.4. INTERFACES DE COMUNICAÇÃO

As interfaces de comunicação entre o cliente e o servidor já encontram-se implementadas na atual infra-estrutura oferecida pelas companhias telefônicas celulares do município do Rio de Janeiro e pelos fabricantes de celulares.

3.4. REQUISITOS DE DESEMPENHO

O sistema deverá estar disponível nas regiões onde haja a cobertura oferecida pela rede de telefonia móvel. Além disso, o desempenho esperado é de que o usuário consigar realizar consultas e cadastros em um período não superior a cinco minutos.

3.5. Restrições de Projeto

Relativo ao dispositivo utilizado pelo cliente, existe uma limitação de área visual para exibir as telas, o aparelho também possui baixa capacidade de processamento e de memória.

3.6. Atributos

Não aplicado

3.7. Outros Requisitos

Não aplicado

1. INTRODUÇÃO

1.1. FINALIDADE DO PLANO

Projetar um sistema móvel de informação e gerenciamento para o Departamento de Combustíveis, Divisão de Preços de uma empresa de petróleo. O plano será realizado por Ingrid Teixeira. O software terá a denominação de *FMMIS – Fuels Marketing Mobile Information System*

1.2. ESCOPO E OBJETIVOS DO PROJETO

1.2.1. ESCOPO

O sistema de suporte a vendas tem o objetivo de fornecer informações unificadas, abrangentes e atualizadas aos funcionários remotos, incluindo situação dos pedidos de desconto de preços, posição dos preços da concorrência e mensagens diversas. Estará disponível através de telefone celular sendo atualizado toda vez que for estabelecida conexão com o computador central da Cia.

A força de vendas está estruturada em filiais, áreas e zonas. Uma filial é responsável pela cobertura de uma área geográfica, estando subdividida em áreas de vendas. Cada área de venda é de responsabilidade de um vendedor titular, que é o gerente de território (GT) da Cia, e a estrutura de visitas é estabelecida de modo que uma área possua várias zonas de vendas e cada zona possua determinado número de clientes a serem visitados em um dia.

1.2.2. PRINCIPAIS FUNÇÕES

GT – Gerente de território

Entrada:

Acesso ao sistema

Efetua pedidos de descontos nos preços dos produtos (perfil GT)

Saída:

Recebe Mensagens sobre Segurança

Consulta os preços dos postos da Cia e da concorrência dentro da(s) FAZ

Consulta o status da solicitação feita (individual)

Consulta avisos da Cia

Consulta avisos do GA

GA – Gerente de área

Entrada:

Acesso ao sistema

Efetua pedidos de descontos nos preços dos produtos (perfil GA)

Cadastra avisos para os GT

Saída:

Recebe Mensagens sobre Segurança

Consulta os preços dos postos da Cia e da concorrência dentro de suas FAZ

Consulta o status das solicitações feitas (individual)

Consulta o status das solicitações feitas por cada GT subordinado

Consulta avisos da Cia

1.2.3. ITENS DE DESEMPENHO

Em ambiente de simulação, onde será utilizada a infra-estrutura da Internet, espera-se uma taxa de transferência de 22kbps.

1.2.4. Gerenciamento e Restrições Técnicas

O plano será desenvolvido por Ingrid Teixeira, tendo a supervisão do professor Antônio Cláudio Gómez Sousa.

2. Estimativas

2.1. Dados Históricos Usados nas Estimativas

Serão utilizados os dados históricos e estatísticos baseados no modelo *CocomoII*.

2.2. Técnicas de Estimativa

Técnica de ponto por função do programa *CocomoII*, utilizando as estimativas de telas de entrada, saída, consultas, baseadas no item 1.2.2. Será utilizado um peso de experiência e capacidade de programação e análise da equipe.

2.3. Estimativas de Esforço, Custo e Duração

O resultado obtido foi de que o esforço será de 15,3. O tempo necessário será de 10 meses de 4 horas cada dia.

3. Estratégia para o Gerenciamento dos Riscos

3.1. Tabela de Riscos

Código	Descrição	Categoria	Probabilidade (%)	Impacto	RMMM
1	D1	Tecnologia	30	1	A
2	D2	Prazo	50	3	B
3	D3	Produto	40	2	C
4	D4	Produto	30	2	D
5	D5	Equipe	30	3	B
6	D6	Cliente	40	2	B
7	D7	Equipe	30	2	D

Consultar tabelas auxiliares 1 e 2.

Tabelas Auxiliares

Tabela 1

D1 – Falta de recursos técnicos adequados: software, hardware e bibliografia

D2 – Atraso na finalização do projeto

D3 – Complexidade excessiva das ferramentas usadas para o desenvolvimento do projeto

D4 – Complexidade excessiva do projeto

D5 – Erro nas estimativas de tempo e do sistema em si

D6 – Novas exigências feitas pelos clientes

D7 – Falta de domínio em relação ao ambiente

Tabela 2

A – Buscar recursos próprios para financiar o projeto

B – Reavaliar o projeto e o prazo

C – Questionar aos fabricantes dessas ferramentas sobre seu uso e/ou buscar alternativas

D – Buscar fontes de informações que capacitem melhor a equipe

3.2. *Discussão dos Riscos a serem Gerenciados*

Risco 1, devido a tecnologia empregada ser recente, há poucos recursos bibliográficos e grande dificuldade para implementar o ambiente de desenvolvimento.

Risco 2, o atraso pode ocorrer dada a complexidade de implementação do ambiente.

Risco 3, essa complexidade é causada pelo estado da arte da tecnologia empregada.

Risco 4, o projeto pode se tornar complexo caso as ferramentas se mostrem inadequadas.

Risco 5, podem existir erros devido à inexperiência da equipe

Risco 6, as exigências podem afetar o prazo, aumentar a complexidade e alterar a lógica da programação

Risco 7, como ainda não existe um ambiente de desenvolvimento de projetos sólido no mercado, a equipe não domina totalmente as ferramentas

3.3. *Plano RMMM para cada Risco*

3.3.1. Como Mitigar os Riscos

Risco 1, buscar os recursos técnicos com antecedência em relação aos prazos estabelecidos. Além disso, deve-se ambientar a equipe com as ferramentas a serem utilizadas, também com antecedência

Risco 2, buscar recursos bibliográficos e técnicos com antecedência. E além disso, fazer testes com todos os *softwares* que serão necessários no decorrer do projeto

Risco 3, utilização de ferramentas alternativas. Caso não haja, deverá ser proposto um novo modelo para o sistema

Risco 4, replanejamento do sistema a fim de reduzir sua complexidade

Risco 5, estudo mais aprofundado do contexto onde o sistema está inserido – ambiente de aplicações sem fio

Risco 6, desenvolver o projeto dentro das especificações indicadas pelos usuários

Risco 7, idem ao risco 5

3.3.2. COMO MONITORAR OS RISCOS

Risco 1, acompanhar periodicamente os recursos técnicos disponíveis

Risco 2, avaliação constante do cronograma de trabalho proposto

Risco 3, verificar a qualidade das ferramentas em uso no decorrer do projeto

Risco 4, questionar a equipe sobre o desenvolvimento do sistema

Risco 5, avaliar a complexidade do sistema periodicamente

Risco 6, avaliação do impacto de novas exigências feitas pelos usuários no desenvolvimento do projeto

Risco 7, acompanhar a adaptação da equipe no ambiente do projeto

3.3.3. COMO GERENCIAR OS RISCOS – PLANO DE CONTINGÊNCIA

Risco 1, busca de fontes de informação que capacitem melhor a equipe

Risco 2, reduzir a complexidade do projeto

Risco 3, buscar alternativas disponíveis no mercado

Risco 4, idem ao risco 1

Risco 5, diminuir o sistema ou aumentar o prazo final

Risco 6, avaliação dos impactos no sistema

Risco 7, idem ao risco 1

4. CRONOGRAMA

4.1. Estrutura do Projeto

Não será feito um *breakdown* do projeto, portanto ele será composto por apenas um módulo.

4.2. Rede de Tarefas

	Nome da Tarefa	*Duração*	*Início*	*Fim*
1	Planejamento e Requisitos	3 meses	01/10/2001	31/12/2001
2	Design do Produto	1 mês	01/01/2002	31/01/2002
3	Programação (Design detalahdo)	3 meses	01/02/2002	31/04/2002
4	Programação (Código e Teste da unidade)	2 mês	01/05/2002	31/06/2002
5	Integração e Testes	1 mês	01/07/2002	31/07/2002

4.3. Tabela de Recursos

Fase	*Recursos*	*Uso*
Análise & Planejamento	MSWord 97 Cocomo II	Geração de toda a documentação Construção do cronograma Obtenção das estimativas de tempo, esforço e tempo de execução
Design do produto	MSPowerPoint 97 Emulador WAP	Geração de *slides* com as principais funcionalidades do sistema Desenho do protótipo
Programação	Emulador WAP PHP MySQL Internet	Design detalhado, código e teste do sistema Banco de Dados Hospedagem WAP
Integração & Testes	Emulador WAP Internet	Detecção de falhas

5. Recursos do Projeto

5.1. Pessoal

Ingrid Teixeira

5.2. Hardware & Softwares

Hardware

Pentium 133Mhz

128Mb RAM

Software

MSWord 97

Cocomo II

MSPowerPoint 97

Emulador WAP

PHP

ASP – Active Sever Page

MySQL

5.3. Recursos Especiais

Internet

6. Organização da Equipe

Todas as tarefas serão realizadas pelo único membro da equipe.

7. Mecanismos de Controle e Acompanhamento

7.1. Controle e Garantia da Qualidade

Estão previstas as seguintes revisões técnicas formais:

8 de novembro – Primeira revisão técnica formal

12 de novembro – Segunda revisão técnica formal

19 de novembro – Terça revisão técnica formal

22 de novembro – Quarta revisão técnica formal

26 de novembro – Quinta revisão técnica formal

7.2. Controle e Gerenciamento das Mudanças

Toda mudança será devidamente documentada de acordo com as novas exigências feitas ao projeto.

8. Apêndices e Anexos

Não aplicável

O CD-ROM que acompanha esta documentação impressa tem seu conteúdo descrito abaixo, na árvore de diretórios.

/apresentacao: Contém em formato PDF a apresentação dos slides da defesa deste projeto.

/documentacao: Contém este documento em formato PDF

/wml: Contém os arquivos WML e PHP do FMMIS

/fmmis.mysql: Contém todos os arquivos necessários à carga e à criação do banco de dados FMMIS, no mysql.