

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO – BACHARELADO**

**APLICATIVO PARA REPRESENTANTE COMERCIAL EM**  
**DISPOSITIVO MÓVEL (PDA) USANDO A TECNOLOGIA**  
**J2ME E BANCO DE DADOS**

**ROBSON LUIS DA SILVA**

**BLUMENAU**  
**2005**

**2005/2-24**

**ROBSON LUIS DA SILVA**

**APLICATIVO PARA REPRESENTANTE COMERCIAL EM  
DISPOSITIVO MÓVEL (PDA) USANDO A TECNOLOGIA  
J2ME E BANCO DE DADOS**

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Ciências  
da Computação — Bacharelado.

Prof. Alexander Roberto Valdameri - Orientador

**BLUMENAU  
2005**

**2005/2-24**

**APLICATIVO PARA REPRESENTANTE COMERCIAL EM  
DISPOSITIVO MÓVEL (PDA) USANDO A TECNOLOGIA  
J2ME E BANCO DE DADOS**

Por

**ROBSON LUIS DA SILVA**

Trabalho aprovado para obtenção dos créditos  
na disciplina de Trabalho de Conclusão de  
Curso II, pela banca examinadora formada  
por:

Presidente: \_\_\_\_\_  
Prof. Alexander Roberto Valdameri – Mestre - Orientador, FURB

Membro: \_\_\_\_\_  
Prof. Francisco Adell Péricas, Mestre - FURB

Membro: \_\_\_\_\_  
Prof. Maurício Capobianco Lopes, Mestre - FURB

Blumenau, 18 de novembro de 2005

Dedico este trabalho aos meus familiares, a minha namorada, aos amigos e mestres da universidade, especialmente aqueles que me ajudaram na realização deste.

## **AGRADECIMENTOS**

Aos meus pais, Gilsemar e Jucélia, pelo incentivo e pelas cobranças, mas acima de tudo por acreditarem que seria possível chegar até aqui.

A minha irmã, Juliana, por sempre estar disposta a me ajudar quando preciso.

A minha namorada, Milena, por ter me dado todo o apoio possível e por ter sido compreensiva durante o desenvolvimento deste trabalho.

Aos amigos que acreditam em mim e torcem por mim.

Ao meu padrinho, Edson Savi, por ter me mostrado, mesmo sem querer ou até mesmo sem saber, o caminho correto a ser seguido por uma pessoa que alcança uma universidade.

Ao meu orientador, Alexander Valdameri, por ter acreditado que seria possível a realização deste trabalho e por toda a ajuda prestada.

Quem perde seus bens, perde muito; quem perde um amigo, perde mais; mas quem perde a coragem, perde tudo.

Miguel de Cervantes

## RESUMO

Este trabalho apresenta uma especificação e desenvolvimento de um protótipo de software para representantes comerciais utilizando dispositivos móveis. A aplicação disponibiliza dados relevantes da empresa para os seus representantes, visando auxiliar a troca de informações entre a empresa e funcionários que trabalham fora dela. O protótipo faz uso da tecnologia Java voltada para estes dispositivos, a especificação *Java Micro Edition* (J2ME), e uso de banco de dados em dispositivos móveis.

Palavras-chave: Representante comercial. Dispositivo móvel. J2ME. Banco de dados.

## **ABSTRACT**

This work presents a specification and development of an archetype of software directed toward commercial representative making use of mobile devices. The application gives relevant data the company for its representatives, aiming at to assist the exchange of information between the company and employees who work outside of it. The application make use of the Java technology directed toward these devices, the specification Java Micro Edition (J2ME), and the use of database in mobile devices.

Key-words: Commercial representative. Mobile device. J2ME. Database.

## LISTA DE ILUSTRAÇÕES

FIGURA 1 – Componentes da plataforma Java .....	18
FIGURA 2 – Componentes J2ME.....	19
QUADRO 1 – Exemplo de arquivo XML.....	24
FIGURA 3 – Casos de uso relacionados a cadastros .....	31
FIGURA 4 – Casos de uso relacionados a consultas .....	32
FIGURA 5 – Casos de uso relacionados a importações e exportações de dados.....	32
FIGURA 6 – Limpeza da base de dados .....	33
FIGURA 7 – Diagrama de atividades do caso de uso “cadastrar cliente” .....	34
FIGURA 8 – Diagrama de atividades do caso de uso “consultar dados do representante” .....	34
FIGURA 9 – Diagrama de atividades do caso de uso “cadastrar pedido” .....	35
FIGURA 10 – Diagrama de atividades do caso de uso “listar clientes” .....	36
FIGURA 11 – Diagrama de atividades do caso de uso “listar faturamento” .....	36
FIGURA 12 – Diagrama de atividades do caso de uso “listar pedido”.....	37
FIGURA 13 – Diagrama de atividades do caso de uso “importar dados” .....	37
FIGURA 14 – Diagrama de atividades do caso de uso “exportar dados”.....	38
FIGURA 15 – Diagrama de atividades do caso de uso “limpar base de dados”.....	38
FIGURA 16 – Diagrama de classes da aplicação.....	39
FIGURA 17 – Diagrama de classes de saída de dados .....	40
FIGURA 18 – Diagrama de classes de entrada de dados .....	41
FIGURA 19 – Modelo entidade relacionamento da base de dados do dispositivo móvel .....	42
FIGURA 20 – Modelo entidade relacionamento da base de dados do servidor.....	43
QUADRO 2 - Codificação do acesso ao banco de dados.....	45
QUADRO 3 - Codificação de consulta ao banco de dados.....	46
QUADRO 4 - Codificação de atualização de dados.....	46
QUADRO 5 - Codificação de conexão e recebimento de dados.....	47
QUADRO 6 - Codificação de componentes de interface gráfica.....	47
FIGURA 21 – Tela inicial .....	48
FIGURA 22 – Menu de opções .....	49
FIGURA 23 – Tela de clientes .....	50
FIGURA 24 – Seleção de clientes cadastrados .....	50
FIGURA 25 – Dados do cliente selecionado .....	51

FIGURA 26 – Cadastramento de novo cliente.....	52
FIGURA 27 – Dados do representante.....	53
FIGURA 28 - Lista de produtos .....	53
FIGURA 29 – Detalhes do produto.....	54
FIGURA 30 – Tela de pedidos .....	54
FIGURA 31 – Inclusão de pedido .....	55
FIGURA 32 – Inclusão de itens do pedido.....	55
FIGURA 33 – Consulta faturamento.....	56
FIGURA 34 – Detalhes da nota fiscal .....	57
FIGURA 35 – Importação de dados .....	57
FIGURA 36 – Exportação de dados .....	58
FIGURA 37 – Limpeza da base de dados .....	59

## LISTA DE SIGLAS

.NET – *dot NET*

API – *Application Programing Interface*

BCC – *Curso de Ciências da Computação – Bacharelado*

CE – *Compact Edition*

CDC – *Connected Device Configuration*

CLDC – *Connected Limited Device Configuration*

CRM – *Customer Relationship Management*

DDL – *Data Definition Language*

HTML – *Hipertext Markup Languge*

HTTP – *Hipertext Transfer Protocol*

J2EE – *Java 2 Enterprise Edition*

J2ME – *Java 2 Micro Edition*

J2SE – *Java 2 Standard Edition*

JAD – *Java Descriptor*

JAR – *Java Archive*

JVM – *Java Virtual Machine*

KVM – *Kilo Virtual Machine*

MIDP – *Mobile Information Device Profile*

PDA – *Personal Digital Assitant*

PDAP – *Personal Digital Assistant Profile*

PRC – *Palm Resource*

RMI – *Remote Method Invocation*

UML – *Unified Modeling Language*

URL – *Unified Resource Locator*

SQL – *Structured Query Language*

XML – *Extensible Markup Language*

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>14</b>
1.1 OBJETIVOS DO TRABALHO .....	15
1.2 ESTRUTURA DO TRABALHO .....	16
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>17</b>
2.1 TECNOLOGIA JAVA .....	17
2.1.1 JAVA MICRO EDITIN – J2ME .....	18
2.1.2 CONFIGURAÇÕES E PERFIS.....	19
2.1.3 Perfil MIDP .....	21
2.1.4 IBM J9 VM .....	22
2.2 SERVLETS JAVA .....	22
2.3 A LINGUAGEM XML .....	23
2.4 PLATAFORMA PALM.....	25
2.5 DB2 EVERYPLACE.....	25
2.6 TRABALHOS CORRELATOS .....	27
<b>3 DESENVOLVIMENTO DO APLICATIVO .....</b>	<b>28</b>
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	28
3.2 REQUISITOS NÃO FUNCIONAIS .....	30
3.3 ESPECIFICAÇÃO .....	30
3.3.1 CASOS DE USO DO PROBLEMA .....	30
3.3.2 DIAGRAMA DE ATIVIDADES DA IMPLEMENTAÇÃO.....	33
3.3.3 DIAGRAMA DE CLASSES DA APLICAÇÃO.....	39
3.3.4 DIAGRAMA DE CLASSES DO SERVIDOR .....	40
3.3.5 MODELO ENTIDADE RELACIONAMENTO .....	41
3.4 IMPLEMENTAÇÃO .....	43
3.4.1 TÉCNICAS E FERRAMENTAS UTILIZADAS.....	43
3.4.1.1 Plataforma Eclipse .....	44
3.4.1.2 J2ME Wireless Toolkit .....	44
3.4.1.3 Palm OS Simulator .....	45
3.4.2 Codificação .....	45
3.4.3 Operacionalidade da implementação .....	48
3.5 RESULTADOS E DISCUSSÃO .....	59

3.5.1 PROBLEMAS ENCONTRADOS DURANTE O DESENVOLVIMENTO .....	61
<b>4 CONCLUSÕES .....</b>	<b>62</b>
4.1 EXTENSÕES .....	63
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>64</b>
<b>APÊNDICE A – Detalhamento dos casos de uso do sistema .....</b>	<b>66</b>
<b>APÊNDICE B – Estrutura dos arquivos XML.....</b>	<b>71</b>

## 1 INTRODUÇÃO

Nos tempos de hoje, pode-se afirmar que é praticamente possível acessar informações de qualquer lugar do planeta, em qualquer momento. Celulares, pagers e palmtops fazem parte do dia-a-dia de uma quantidade cada vez maior de pessoas.

Segundo Mateus (1998, p. 11), independente do tipo de dispositivo móvel, a maior parte desses equipamentos deve ter capacidade de se comunicar com a parte fixa da rede e possivelmente com outros computadores móveis. Fazendo uma análise dos computadores atuais, estes ficam residentes em um lugar fixo, ou seja, não tem esta característica de mobilidade.

Comparando-se um computador portátil com um computador de mesa, os modelos portáteis são pequenos e alimentados por bateria, possuem um pequeno visor e pouca memória disponível, já os computadores convencionais possuem um alto poder de processamento, muita memória disponível e muita capacidade de armazenamento.

Os dispositivos móveis têm tido o seu poder de processamento aumentado muito rapidamente nos últimos anos. Alguns modelos, de acordo com PalmOne (2005) já possuem processadores com 416 MHz e conforme Dell (2005), o modelo Axim X30 possui processador com 624 MHz e sistema operacional Microsoft Windows Mobile, voltado para este tipo de dispositivo.

É inegável que estes dispositivos, no futuro, terão tanto poder de processamento quanto um computador convencional, mas tendo a vantagem da mobilidade, podendo acessar qualquer informação de qualquer lugar a qualquer momento.

Com um dispositivo móvel, por exemplo, um representante pode visitar seus clientes e ter à mão toda a posição do mesmo. Faturamento, produtos mais comprados, emissão de novos pedidos são alguns exemplos dos dados que podem ser trabalhados junto ao cliente

utilizando-se apenas de um dispositivo móvel.

Este trabalho tem como objetivo tornar realidade o uso da computação móvel dentro de uma empresa metalúrgica, desenvolvendo um aplicativo para uso em dispositivos móveis conhecidos como *Personal Digital Assistant* (PDA).

O aplicativo disponibiliza dados para os representantes da empresa, fornecendo dados da sua carteira de pedidos, clientes e estatísticas relacionadas. A empresa já conta com um sistema integrado desenvolvido com a tecnologia *Java 2 Standard Edition* (J2SE).

Este sistema integrado abrange todas as áreas internas da empresa, tais como faturamento, contas a pagar e receber e produção. O sistema possui a base de dados em um banco de dados relacional *PostgreSQL*.

Neste contexto, o trabalho utiliza a mesma tecnologia Java, mas voltada para dispositivos móveis, a tecnologia *Java 2 Micro Edition* (J2ME).

Seguindo este pensamento o aplicativo faz uso de um banco de dados relacional para armazenar as informações recebidas do servidor da empresa. O banco de dados da empresa não é o mesmo do dispositivo móvel, limitando a conexão entre o aplicativo móvel e o banco de dados da empresa. Como solução a comunicação entre a empresa e o dispositivo móvel sempre se dá através de troca de arquivos com o padrão *Extensive Markup Language* (XML).

Conforme Deitel (2001, p. 7), a linguagem XML fornece a estrutura ideal para a troca de informações entre sistemas e plataformas distintas, oferecendo então uma solução para a troca de informações entre a empresa e o dispositivo móvel.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver um aplicativo para dispositivos móveis (PDA), usando a tecnologia J2ME, aliada ao uso de bancos de dados, visando atender às necessidades

existentes hoje em uma empresa metalúrgica. O sistema desenvolvido manterá no dispositivo móvel dados de clientes, faturamento e permitirá ao usuário entrar com dados, neste caso, pedidos de vendas.

Os objetivos específicos do trabalho são:

- a) disponibilizar dados de clientes e situação da carteira de vendas para usuários que utilizam *palmtop*;
- b) importar dados disponibilizados pela empresa através do uso da linguagem XML;
- c) permitir alterações dos dados no dispositivo móvel;
- d) disponibilizar dados do dispositivo móvel para a empresa através do uso da linguagem XML;
- e) utilizar banco de dados no dispositivo móvel.

## 1.2 ESTRUTURA DO TRABALHO

A estrutura deste trabalho está dividida em quatro capítulos.

O segundo capítulo apresenta a tecnologia utilizada para o desenvolvimento do protótipo, discutindo os pontos em que levam ao uso da especificação *Java Micro Edition* como solução para o problema apresentado. Este capítulo ainda apresenta uma visão geral sobre banco de dados usado e o uso em dispositivos móveis, a linguagem XML e o seu uso em desenvolvimento de sistemas.

O terceiro capítulo apresenta o desenvolvimento do protótipo, sua especificação, ferramentas utilizadas e sua implementação.

O quarto capítulo apresenta os resultados, discussões e extensões do trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 TECNOLOGIA JAVA

Segundo Sun (2005), a tecnologia Java define uma linguagem de programação e uma plataforma de software, cujo principal objetivo é a portabilidade do código, isto é, o mesmo programa pode ser executado em sistemas operacionais diferentes.

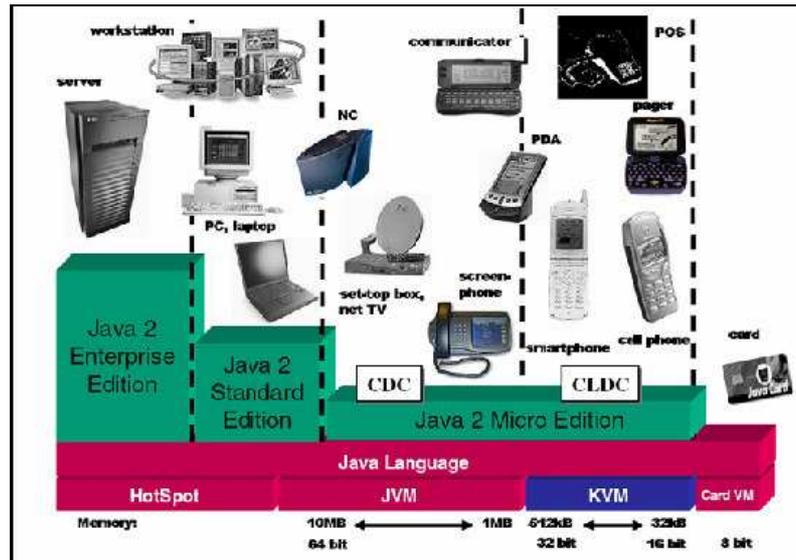
A linguagem de programação Java permite que sejam criados programas para serem executados em desktops, servidores e diversos outros dispositivos entre eles PDAs e celulares.

Conforme Schmitt Junior (2004), os programas criados com a linguagem Java não geram códigos nativos do sistema operacional, ao invés disso geram código compilado, chamado de *bytecode*, que são interpretados através de outro programa chamado *Java Virtual Machine* (JVM), este por sua vez, interpreta o *bytecode* e gera o código binário para o sistema operacional onde está sendo executado o programa.

De acordo com Sun (2005), a plataforma Java é dividida em três edições:

- a) *Java 2 Platform, Standard Edition*: possibilita a construção de aplicações cliente;
- b) *Java 2 Platform, Enterprise Edition*: possibilita a construção de aplicações para servidores;
- c) *Java 2 Platform, Micro Edition*: possibilita a construção de aplicações voltadas para dispositivos móveis como celulares e *handhelds*.

A figura 1 mostra como está dividida a plataforma Java.



Fonte: Corbera (2003)

FIGURA 1 – Componentes da plataforma Java

### 2.1.1 JAVA MICRO EDITIN – J2ME

De acordo com Sun (2005), *Java 2 Micro Edition (J2ME)*, é um padrão que provê um ambiente flexível para aplicações que executem em dispositivos como celulares, PDAs, TVs, sistemas de navegação de carros e outros dispositivos.

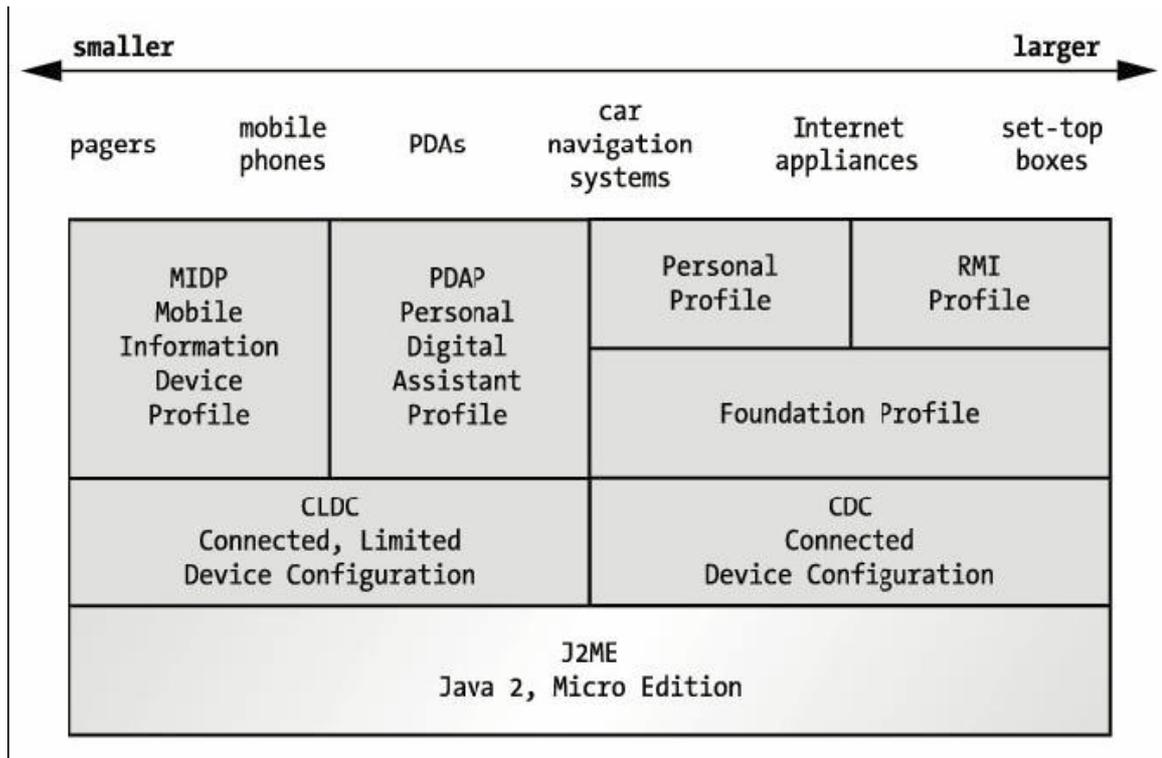
Basicamente, J2ME, é uma versão compacta da especificação J2SE e oferece uma máquina virtual voltada para os dispositivos móveis, chamada de *Kilobyte Virtual Machine (KVM)*.

Conforme Fonseca (2002, p. 28), KVM é uma implementação da máquina virtual otimizada para uso em dispositivos de recursos limitados, que aceita o mesmo conjunto de *bytecodes* que a máquina virtual clássica aceita. A letra “K” vem de “Kilo”, assim chamada porque a sua utilização de memória pode ser medida em *kilobytes*.

De acordo com Schmitt Junior (2004), vários dispositivos diferentes têm características em comum, mas cada dispositivo é construído para o seu propósito, tendo assim

características diferenciadas entre eles a fim de atender o seu propósito. Desta forma, J2ME dispõe de um conjunto de componentes para atender as diferenças entre os dispositivos, divididos em perfis (*profiles*), configurações (*configurations*) e pacotes extras.

A figura 2 mostra os componentes que formam a J2ME.



Fonte: Knudsen (2003)

FIGURA 2 – Componentes J2ME

### 2.1.2 CONFIGURAÇÕES E PERFIS

As configurações do J2ME definem uma plataforma Java para uma determinada faixa de dispositivos, definidos por uma série de características como:

- informações sobre a memória do dispositivo;
- tipo e velocidade do processador;

c) conectividade do aparelho.

Há duas configurações disponíveis para o J2ME:

- a) *Connected Limited Device Configuration* (CLDC), voltada para dispositivos com recursos limitados;
- b) *Connected Device Configuration* (CDC), voltada para dispositivos com um poder de processamento maior, conectividade e maior recurso de memória.

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

Conforme Schmitt Junior (2004), alguns perfis associados à configuração CDC são :

- a) *Foundation Profile*;
- b) *Personal Profile*;
- c) *Remote Method Invocation (RMI) Profile*;
- d) *Personal Basis Profile*.

São perfis associados à configuração CLDC:

- a) *Mobile Information Device Profile* (MIDP);
- b) *Personal Digital Assistant Profile* (PDAP).

Conforme Sun (2005), no começo do ano de 2003 a configuração CLDC recebeu uma atualização, chegando à versão 1.1. Os objetivos desta versão eram os mesmos objetivos iniciais, atingir dispositivos com recursos limitados, porém adicionando novos recursos como suporte a números com ponto flutuante.

### 2.1.3 Perfil MIDP

De acordo com Wilding-McBride (2003, p. 31), a especificação do perfil MIDP é voltada para dispositivos que suportem a configuração CLDC. Estes dispositivos normalmente são telefones celulares e PDAs.

De acordo com a especificação do perfil MIDP, o dispositivo deve ter pelo menos 128 kb de memória não volátil para o armazenamento da implementação, 32 kb de memória volátil para a pilha Java e 8 kb de memória não volátil para armazenamento persistente.

O dispositivo deve ainda ter algum recurso de entrada de dados como um teclado, tela por toque ou área de teclas como telefones celulares. É necessário ainda a possibilidade de conexão a algum tipo de rede e um visor de pelo menos 96 x 54 pixels.

Confirme Sun (2005), a o perfil MIDP está dividido em duas versões. A primeira versão, MIDP 1.0, define todo o conjunto de bibliotecas requeridas pelas aplicações voltadas para dispositivos móveis associados à configuração CLDC. A segunda versão, MIDP 2.0, é uma atualização da especificação MIDP, com recursos novos e melhorados. Alguns recursos da especificação MIDP 2.0 são:

- a) interface com usuário melhorada;
- b) suporte a funções multimídia;
- c) suporte a jogos;
- d) suporte a novos tipos de conexão, como HTTP;
- e) novas especificações de segurança.

Conforme Schmitt Junior (2004, p. 23), uma aplicação MIDP contém no mínimo uma classe MIDlet, podendo em alguns casos conter mais de um MIDlet, sendo chamada de *MIDlet Suite*. A aplicação MIDP, para ser instalada no dispositivo, precisa ser empacotada,

criando um *Java Archive* (JAR) que obrigatoriamente contém um arquivo de manifesto, englobando as informações sobre o MIDlet contido no pacote JAR. Este arquivo contendo as informações sobre a aplicação, configuração e perfil utilizado na mesma, é denominado *Java Application Descriptor* (JAD).

De acordo com Muchow (2001, p. 444), para que possa ser executado no Palm OS, este par de arquivos deve ser convertido em um único arquivo denominado *Palm Resource* (PRC). Segundo Rodhes (1998, p. 89), uma aplicação destinada ao sistema operacional Palm é armazenada em forma de recursos, contendo o código da aplicação, interfaces, ícones, textos e etc.

#### 2.1.4 IBM J9 VM

O projeto ao qual este trabalho é destinado necessita da utilização de elementos que estão presentes na configuração CLDC 1.1. A máquina virtual Java disponibilizada pela Sun Microsystems para o sistema operacional Palm não é compatível com esta configuração, por isso é necessário o uso de uma máquina virtual que suporte esta configuração.

Conforme IBM (2005), a máquina virtual Java desenvolvida pela IBM é denominada J9. Esta máquina virtual possui algumas vantagens sobre a máquina virtual disponibilizada pela Sun Microsystems, entre elas o suporte a CLDC 1.1, MIDP 2.0 e outras configurações e perfis não disponibilizados pela Sun para a plataforma Palm.

## 2.2 SERVLETS JAVA

Segundo Schmitt Junior (2004, p.24), *servlet java* é uma aplicação Java semelhante a uma página da internet, para ser executada num servidor e atender a requisições web. O

servlet recebe a requisição com seus parâmetros e trata a requisição de diversas formas.

O acesso ao servlet pode ser executado através de um endereço *Hipertext Transfer Protocol* (HTTP). Os parâmetros podem ser enviados juntamente com o endereço *Universal Resource Locator* (URL) executado. O resultado do processamento do *servlet* é dado em informações literais, podendo ser uma página internet ou dados no formato XML.

Neste trabalho, foram desenvolvidos *servlets* para tratar as requisições do dispositivo móvel. Estas requisições são responsáveis pela troca de dados entre o dispositivo móvel e a base de dados da empresa. O dispositivo móvel pode fazer acesso a dois tipos de servlet, o primeiro tipo ao ser requisitado, acessa a base de dados da empresa e retorna os dados para o dispositivo através de um arquivo no formato XML. O segundo tipo recebe a requisição do dispositivo móvel e grava os dados, passados pelos parâmetros, na base de dados da empresa.

### 2.3 A LINGUAGEM XML

Segundo Furgeri (2001), em 1996, vários especialistas em *Standard Generalized Markup Language* (SGML), principal linguagem de marcação responsável pelo surgimento do *Hyper Text Markup Language* (HTML), sob a supervisão de Jon Bosak, da Sun Microsystems, reuniram-se para criação de um novo padrão de marcação voltado para a Internet. O objetivo principal era dar suporte a desenvolvedores para a criação de seus próprios marcadores e atributos, e assim não ficarem restritos somente ao HTML.

As características iniciais desejadas para a XML eram definidas em três partes: a linguagem em si (XML-LANG), a ligação entre documentos (XML-LINK) e a forma de apresentação de documentos (XS).

Algumas das principais características desejáveis para a implementação da linguagem na *web* eram as seguintes:

- a) criar uma linguagem simples, que possibilitasse a rápida construção de documentos para uso na *web*;
- b) fornecer suporte à criação de aplicações compatíveis com a abordagem HTML;
- c) possibilitar o desenvolvimento de uma grande variedade de aplicativos, aproveitando-se de seus recursos;
- d) fornecer suporte para a criação de marcadores personalizados, definidos pelo desenvolvedor do documento *web*;
- e) permitir a criação de documentos que pudessem ser validados, isto é que existisse uma forma de verificar a estrutura do documento, verificando se seus elementos eram válidos, da mesma forma que ocorria com a SGML;
- f) fornecer suporte para a criação de *hiperlinks* que fossem compatíveis com a especificação de endereços *Uniform Resource Locator* (URL), criando ligações entre documentos.

Com essas características, a XML oferece um meio completo para a elaboração e distribuição de documentos, sendo independente de plataformas e de sistemas. O objetivo era transformar o conceito da HTML, fornecendo à XML recursos adicionais para a distribuição de documentos.

O quadro 1 mostra um exemplo de arquivo XML.

```
<?xml version="1.0"?>
<contabilidade>
  <lancamento>
    <data>20020101</data>
    <historico>Vendas de mercadorias a prazo</histórico>
    <conta id="4321" acao="C"/>
    <valor>1550,00<valor>
  </lancamento>
</contabilidade>
```

Fonte: Kracik (2002, p. 17)

#### QUADRO 1 – Exemplo de arquivo XML

Arquivos XML podem estar relacionados a outros arquivos, chamados de *Document Type Definition* (DTD), que descreve toda a estrutura de um arquivo XML, suas tags

possíveis, seus atributos e suas combinações. Ao final deste trabalho, no apêndice B, são demonstrados os arquivos que definem as estruturas dos arquivos XML usados na aplicação desenvolvida.

## 2.4 PLATAFORMA PALM

De acordo com PalmBrasil (2005), em 1996 chegou ao mercado um computador de mão chamado Pilot, criado por Jeff Hawkins.

Palm podem ser definidos como computadores de mão ou assistentes pessoais, daí a sigla PDA (*Personal Digital Assistant*), que tem como funções básicas os programas para cadastros de endereços, controle de tarefas, agenda, bloco de notas e e-mail.

Outra característica dos computadores de mão Palm é o reconhecimento de escrita pela tela do aparelho, através do uso de uma caneta disponível com o aparelho.

Segundo PalmSource (2005), o sistema operacional que equipa os aparelhos Palm é conhecido como Palm OS e está presente desde o seu lançamento em 1996.

O sistema operacional foi licenciado por outras empresas que fabricam diferentes aparelhos com o sistema operacional Palm, entre elas estão a Sony e Handspring, que fabricam telefones celulares.

## 2.5 DB2 EVERYPLACE

De acordo com Goya (2004), o IBM DB2 Everyplace é um banco de dados disponível para as plataformas disponíveis em dispositivos móveis, como Palm OS e Windows CE. É considerado o menor banco de dados do mundo ocupando cerca de 100 Kb no dispositivo.

Os dados nos dispositivos móveis podem ser sincronizados com bancos de dados que

estejam em um servidor, inclusive mesmo não sendo um banco de dados IBM DB2, ou seja, podem fazer o sincronismo com banco de dados da Microsoft ou Oracle se for preciso.

Para o uso do IBM DB2 Everyplace no dispositivo móvel, está disponível um conjunto de classes de acesso ao banco de dados. Este conjunto de classes, conhecido por *Java Database Connectivity* (JDBC) pode ser adicionado à aplicação J2ME, possibilitando o acesso ao banco de dados no dispositivo móvel.

Algumas funcionalidades de banco de dados não estão incluídas na versão para dispositivos móveis. São elas:

- a) subconsultas;
- b) criação de visões;
- c) criação de *triggers*;
- d) criação de *stored procedures*;
- e) criação de funções definidas pelo usuário.

Contudo o uso de relacionamento entre tabelas, utilizando chaves primárias e chaves estrangeiras, é o mesmo de um banco de dados voltado para servidores. O uso de comandos DDL para criar e excluir tabelas, índices e outros objetos também são semelhantes a bancos de dados que não são voltados para esta plataforma.

Alguns limites do DB2 Everyplace:

- a) número máximo de índices em uma tabela: 15;
- b) número máximo de chaves estrangeiras em uma tabela: 8;
- c) número máximo de colunas em um índice: 8;
- d) número máximo de colunas em uma chave primária: 8;
- e) número máximo de conexões ao data warehouse: 1;
- f) número máximo de linhas em uma tabela: limitado pelo tamanho da tabela;
- g) número máximo de colunas em uma tabela: 128;

- h) tamanho máximo de uma coluna CHAR: 32 Kb;
- i) tamanho máximo de uma coluna VARCHAR ou BLOB: 32 Kb;
- j) tamanho máximo de inteiro: 31 dígitos.

## 2.6 TRABALHOS CORRELATOS

Em Depiné (2002), é criado um protótipo de software para dispositivos móveis utilizando J2ME para cálculo de regularidade em *rally*. O trabalho cita as principais características de dispositivos móveis, suas vantagens e desvantagens. Este trabalho traz uma explicação detalhada sobre a tecnologia J2ME e suas aplicações.

Em Schaefer (2004), é elaborado um protótipo de aplicativo para transmissão de dados a partir de dispositivos móveis aplicado a uma empresa de transportes. O trabalho traz uma explicação ampla sobre a tecnologia J2ME e toda a sua estrutura. Mostra o uso de dispositivos móveis, neste caso, celulares para transmissão dos dados para a empresa de transportes.

Em Kracik (2002), é elaborada uma proposta para padrão de comunicação e armazenamento de dados contábeis usando XML. O trabalho desenvolvido mostra o uso da tecnologia XML e toda sua estrutura.

Em Galvin (2004), é criado um protótipo de sistema para dispositivos móveis utilizando a plataforma .NET, visando auxiliar o trabalho de relacionamento entre empresa e clientes diretamente no campo, disponibilizando dados para o usuário sobre os seus clientes. O trabalho mostra o uso de dispositivos móveis do tipo *Pocket PC* e *Windows .NET CE*.

Em Schmitt Junior (2004), é desenvolvido um protótipo de *front end* de controle de acesso utilizando J2ME visando uma comunicação com um servidor e troca de dados através de *servlet java*. O trabalho demonstra o uso de componentes da J2ME para persistência de dados.

### 3 DESENVOLVIMENTO DO APLICATIVO

O resultado deste trabalho é um protótipo de uma aplicação voltada para dispositivos móveis que utilizam o sistema operacional Palm e que possuem suporte a J2ME, com o perfil MIDP 2 e configuração CLDC 1.1.

O protótipo combina o uso da tecnologia Java com banco de dados relacionais, que fazem a persistência de dados no dispositivo móvel.

O trabalho faz uso de uma aplicação web que atende e trata todas as requisições do sistema.

Para o desenvolvimento do trabalho foi feito um levantamento e análise definindo as características que o sistema deve ter. A especificação dos requisitos é tratada a seguir.

#### 3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

Os requisitos funcionais declaram e detalham quais são as principais características do sistema com relação a camada de negócios. São requisitos funcionais do sistema:

- a) gravação de dados de clientes digitados pelo usuário;
- b) possibilidade de consulta e gravação dos dados de representantes;
- c) gravação de pedidos digitados pelo usuário;
- d) possibilidade de consulta aos clientes armazenados;
- e) possibilidade de consulta ao faturamento dos clientes;
- f) possibilidade de listar os pedidos cadastrados;
- g) importação de dados dos clientes, produtos e faturamento;
- h) exportação de dados cliente e pedidos digitados para a empresa;
- i) possibilidade de limpar a base de dados do dispositivo móvel.

O sistema disponibilizará um menu principal onde o usuário terá acesso às outras opções do sistema.

A opção de importação deve carregar os dados para o dispositivo móvel através de conexão com o servidor da empresa que disponibilizará os dados.

A opção de exportação deve enviar os dados de pedidos cadastrados e clientes cadastrados para o servidor da empresa, fazendo uma conexão com o mesmo.

A opção de consulta ao faturamento deve permitir ao usuário escolher um cliente na sua base de dados e mostrar todas as notas fiscais emitidas para este cliente. Deve permitir também ao usuário visualizar os detalhes da nota fiscal, isto é, os itens que foram faturados na mesma.

A consulta aos produtos deve disponibilizar ao usuário uma lista de todos os produtos armazenados na base de dados do dispositivo móvel.

Para o cadastramento dos pedidos, o sistema deve permitir a escolha de um cliente e deve sugerir um número de pedido válido, que não esteja na base de dados do *palmtop*. O sistema deve disponibilizar uma interface para o usuário entrar com os itens do pedido, escolhendo o produto através de uma lista dos produtos que estão armazenados na base de dados. Todos os pedidos digitados devem ser marcados para envio para a base de dados do servidor da empresa.

A limpeza da base de dados do dispositivo móvel deve excluir e recriar todas as tabelas do sistema.

### 3.2 REQUISITOS NÃO FUNCIONAIS

São requisitos não funcionais do sistema:

- a) compatibilidade com a plataforma *Palm OS*;
- b) utilização da configuração CLDC 1.1
- c) compatibilidade com recursos do perfil MIDP 2 do J2ME;

### 3.3 ESPECIFICAÇÃO

Para definir a modelagem do sistema foi utilizado a *Unified Modeling Language* (UML), que define diversos diagramas que definem um sistema.

Foram definidos os casos de uso do sistema, diagrama de atividades, diagrama de classes e modelo entidade relacionamento. Todos os diagramas são tratados a seguir.

#### 3.3.1 CASOS DE USO DO PROBLEMA

Os casos de uso são usados para representar a interação do usuário com o sistema, destacando as ações que serão realizadas. No protótipo deste trabalho foram definidos nove casos de uso:

- a) cadastrar cliente;
- b) consultar dados do representante;
- c) cadastrar pedido;
- d) listar produtos;
- e) listar clientes;

- f) listar faturamento;
- g) listar pedidos;
- h) importar dados;
- i) exportar dados;
- j) limpar base de dados.

Os casos de uso relacionados aos cadastros são os cadastros de clientes e cadastros de pedidos. A figura 3 mostra os casos de uso relacionados aos cadastros que podem ser feitos no protótipo.

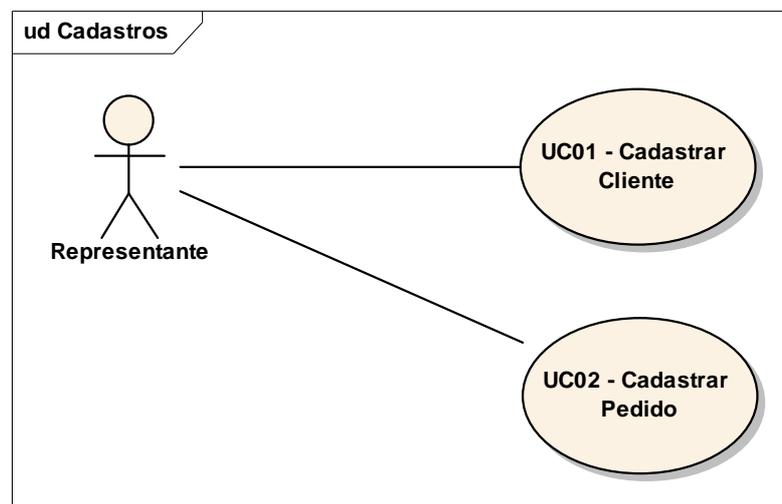


FIGURA 3 – Casos de uso relacionados a cadastros

O aplicativo dispõe de quatro casos de uso relacionados às consultas que o usuário pode fazer. A figura 4 mostra os casos de uso relacionados às consultas que podem ser feitos no protótipo.

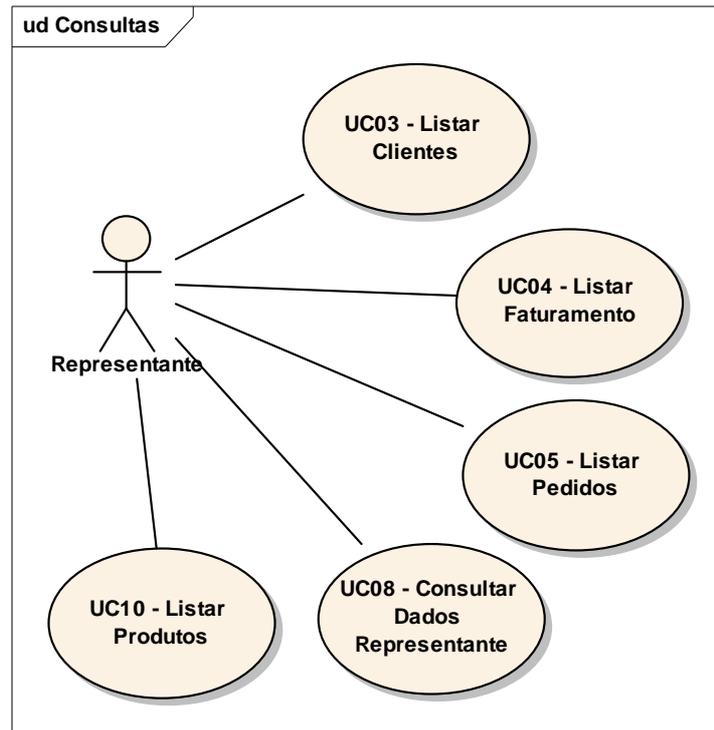


FIGURA 4 – Casos de uso relacionados a consultas

As importações e exportações fazem a troca de dados entre o dispositivo móvel e a base de dados do servidor da empresa. A figura 5 mostra os casos de uso relacionados às importações e exportações de dados que podem ser feitos no protótipo.

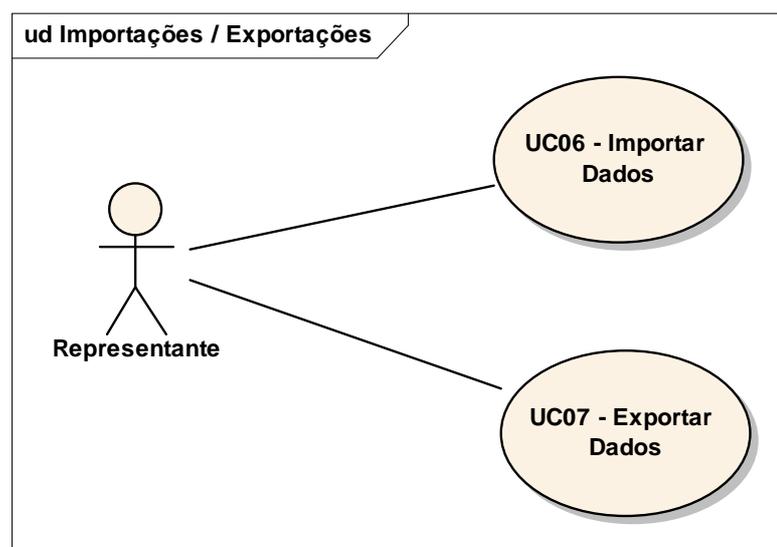


FIGURA 5 – Casos de uso relacionados a importações e exportações de dados

O protótipo disponibiliza uma opção para que o usuário elimine todos os dados da sua base de dados local. A figura 6 mostra o caso de uso relacionado à limpeza de base de dados do dispositivo móvel.

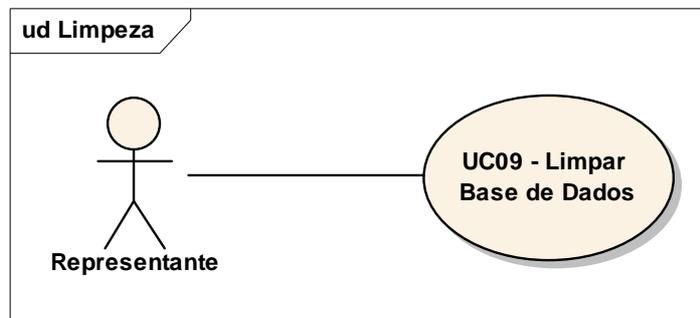


FIGURA 6 – Limpeza da base de dados

Uma descrição detalhada dos casos de uso apresentados aqui pode ser visualizada no apêndice A.

### 3.3.2 DIAGRAMA DE ATIVIDADES DA IMPLEMENTAÇÃO

Os diagramas de atividades servem para ilustrar o fluxo das funções ou passos de cada caso de uso. A seguir, nas figuras 7 a 16, são mostrados os diagramas de atividades relacionados aos casos de uso descritos no trabalho.

A figura 7 mostra o diagrama de atividades do caso de uso relacionado ao cadastramento de clientes.

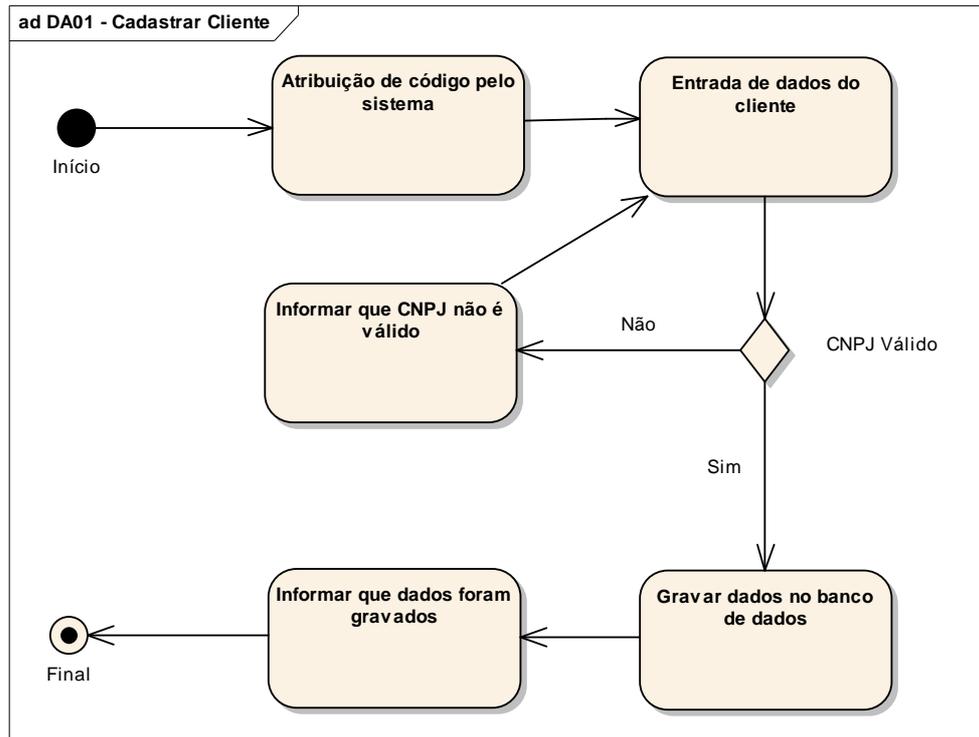


FIGURA 7 – Diagrama de atividades do caso de uso “cadastrar cliente”

A figura 8 mostra o diagrama de atividades do caso de uso que faz a consulta dos dados do representante.

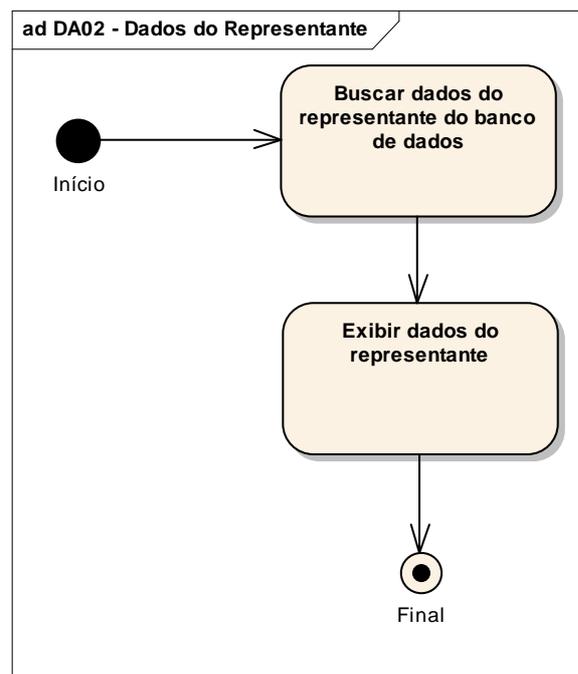


FIGURA 8 – Diagrama de atividades do caso de uso “consultar dados do representante”

A figura 9 mostra o diagrama de atividades do caso de uso relacionado ao cadastramento de pedidos no dispositivo móvel. A aplicação busca os clientes na base de dados, o usuário seleciona o cliente e o sistema sugere um número de pedido. Após isso o usuário faz a entrada de dados dos produtos que compõem o pedido e finaliza.

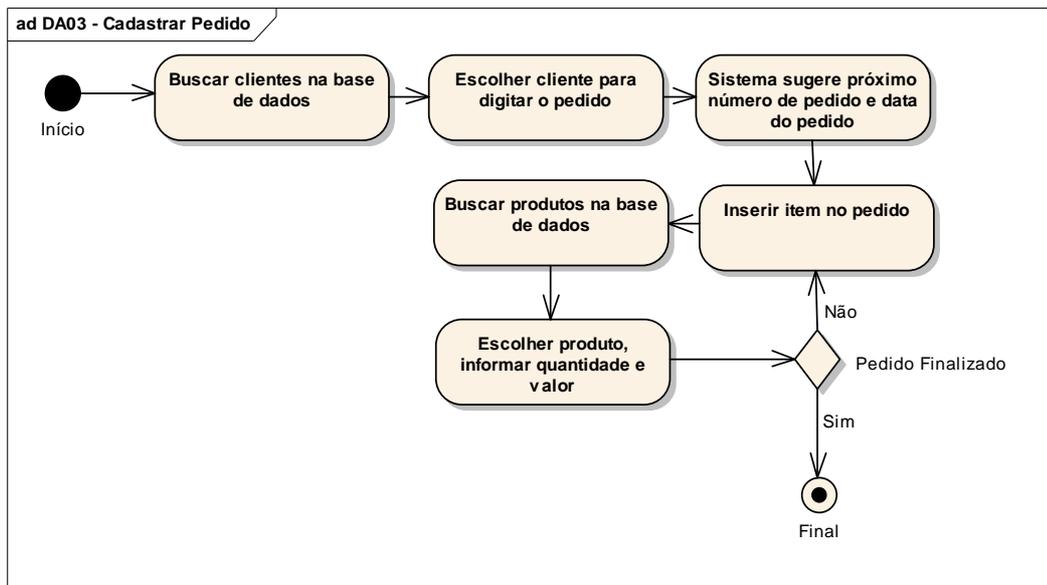


FIGURA 9 – Diagrama de atividades do caso de uso “cadastrar pedido”

A figura 10 mostra o diagrama de atividades do caso de uso listar produtos, onde a aplicação faz acesso à base de dados do dispositivo móvel e exibe uma tela com todos os produtos cadastrados. O usuário pode também visualizar os detalhes de cada produto.

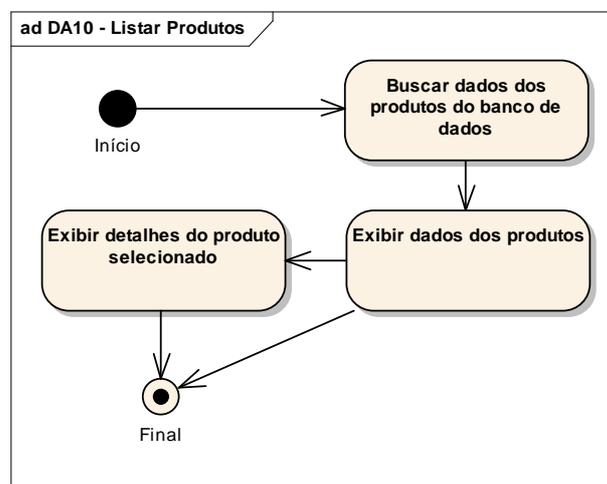


FIGURA 10 – Diagrama de atividades do caso de uso “listar produtos”

A figura 11 mostra o diagrama de atividades do caso de uso listar clientes, onde a aplicação exibe uma tela para o usuário com todos os clientes cadastrados na base de dados do dispositivo móvel.

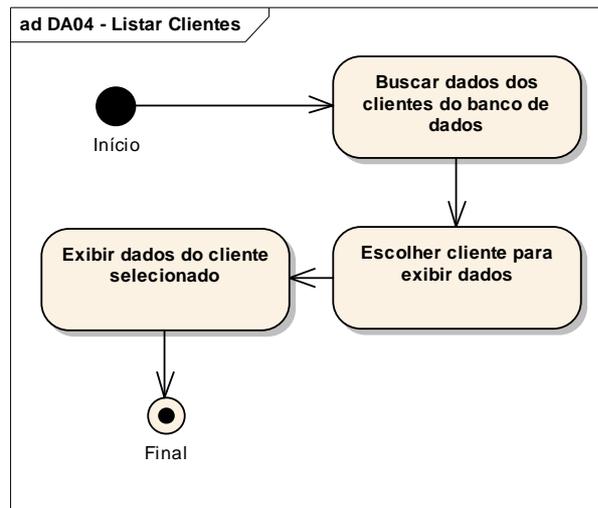


FIGURA 11 – Diagrama de atividades do caso de uso “listar clientes”

A figura 12 mostra o diagrama de atividades relacionado ao caso de uso listar faturamento. A aplicação busca todos os clientes na base de dados e após o usuário escolher um cliente mostra os dados do faturamento do cliente selecionado.

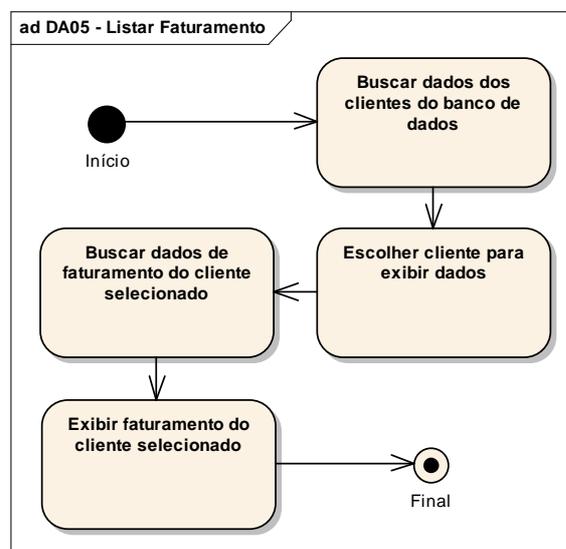


FIGURA 12 – Diagrama de atividades do caso de uso “listar faturamento”

A figura 13 mostra o diagrama de atividades do caso de uso listar pedidos, onde a aplicação faz acesso à base de dados do dispositivo móvel e traz os clientes cadastrados, o usuário escolhe o cliente e a aplicação traz os dados dos pedidos que estão na base de dados.

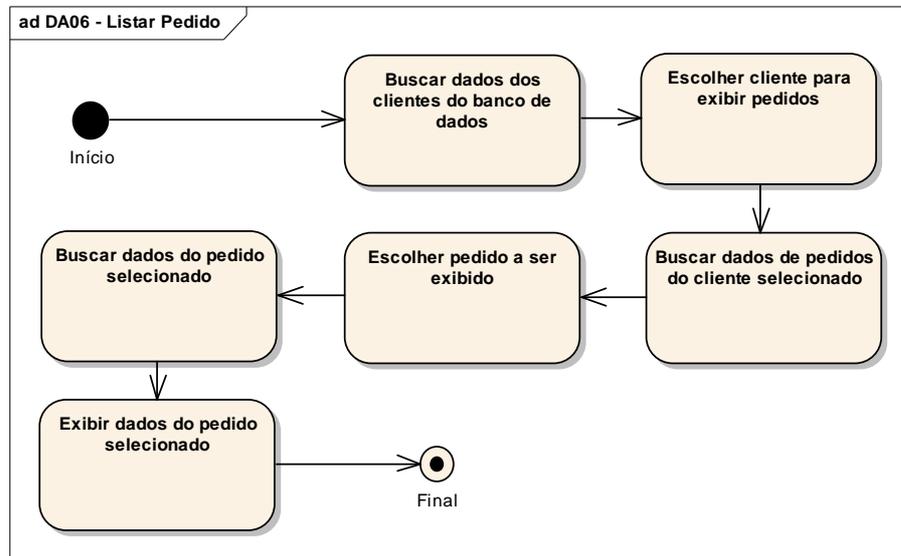


FIGURA 13 – Diagrama de atividades do caso de uso “listar pedido”

A figura 14 mostra o diagrama de atividades do caso de uso importar dados, onde a aplicação faz um acesso ao servidor da empresa, recebe um arquivo no formato XML e trata o arquivo gravando os dados recebidos na base de dados do dispositivo móvel.

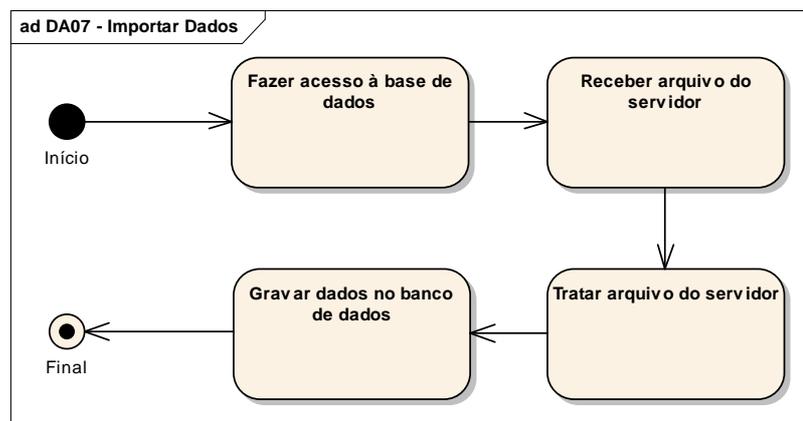


FIGURA 14 – Diagrama de atividades do caso de uso “importar dados”

A figura 15 mostra o diagrama de atividades do caso de uso exportar dados, onde a

aplicação faz um acesso ao servidor da empresa e envia os dados que são tratados pelos *servlets* que ficam no servidor. Os dados são gravados na base de dados do servidor na empresa.

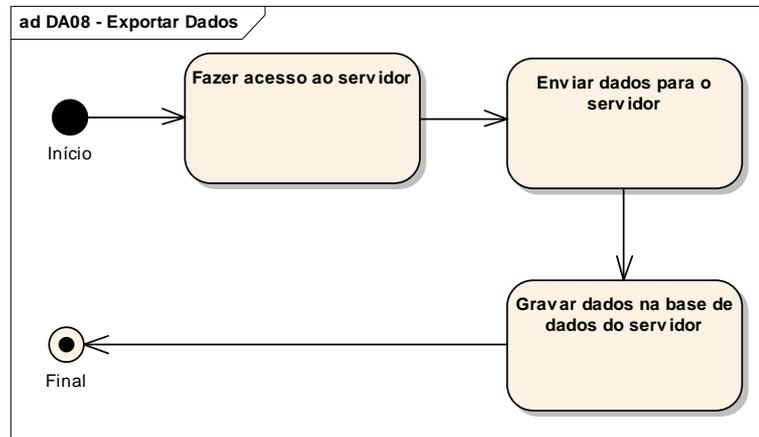


FIGURA 15 – Diagrama de atividades do caso de uso “exportar dados”

A figura 16 mostra o diagrama de atividades do caso de uso limpar base de dados, onde a aplicação faz acesso ao banco de dados do dispositivo móvel e recria as tabelas utilizadas pelo sistema.

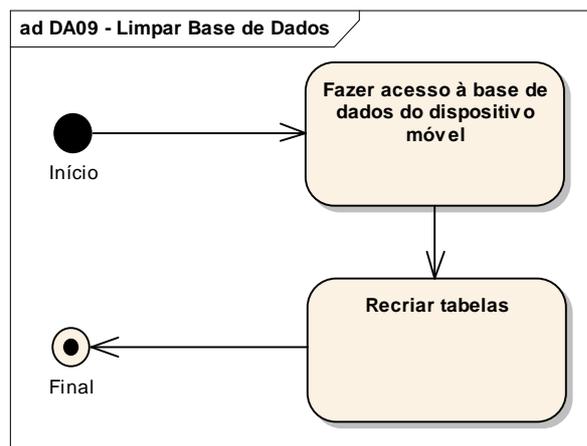


FIGURA 16 – Diagrama de atividades do caso de uso “limpar base de dados”

### 3.3.3 DIAGRAMA DE CLASSES DA APLICAÇÃO

A seguir é demonstrado o diagrama com o conjunto de classes da aplicação, ilustrando o relacionamento entre elas e a sua estrutura simplificada.

A figura 17 mostra as classes da aplicação desenvolvida para o dispositivo móvel.

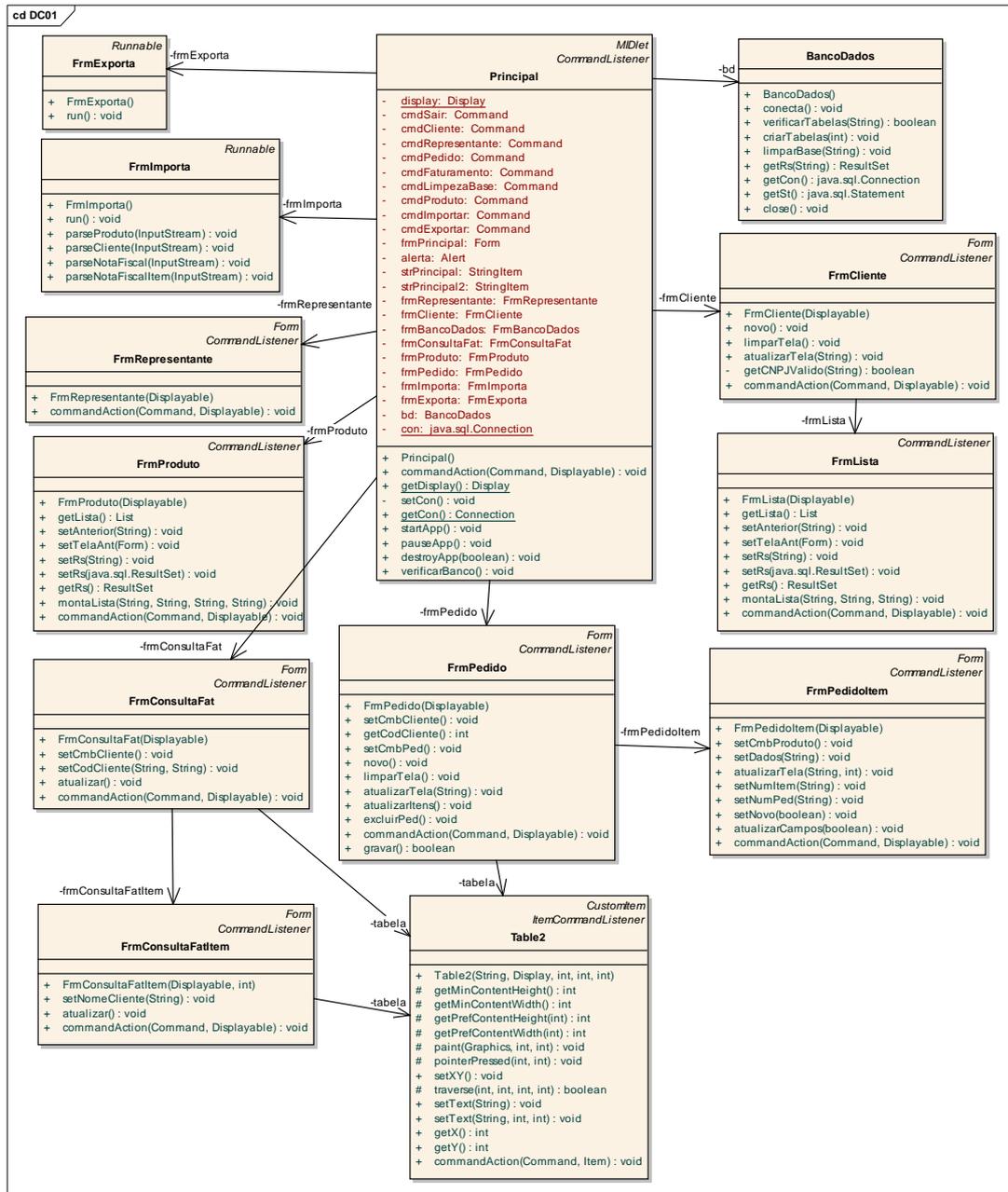


FIGURA 17 – Diagrama de classes da aplicação

### 3.3.4 DIAGRAMA DE CLASSES DO SERVIDOR

A seguir é mostrado o diagrama de classes dos *servlets* desenvolvidos e que ficam no servidor da empresa para atender as requisições do dispositivo móvel.

A figura 18 mostra as classes que tratam as saídas dos dados do servidor para o dispositivo móvel.

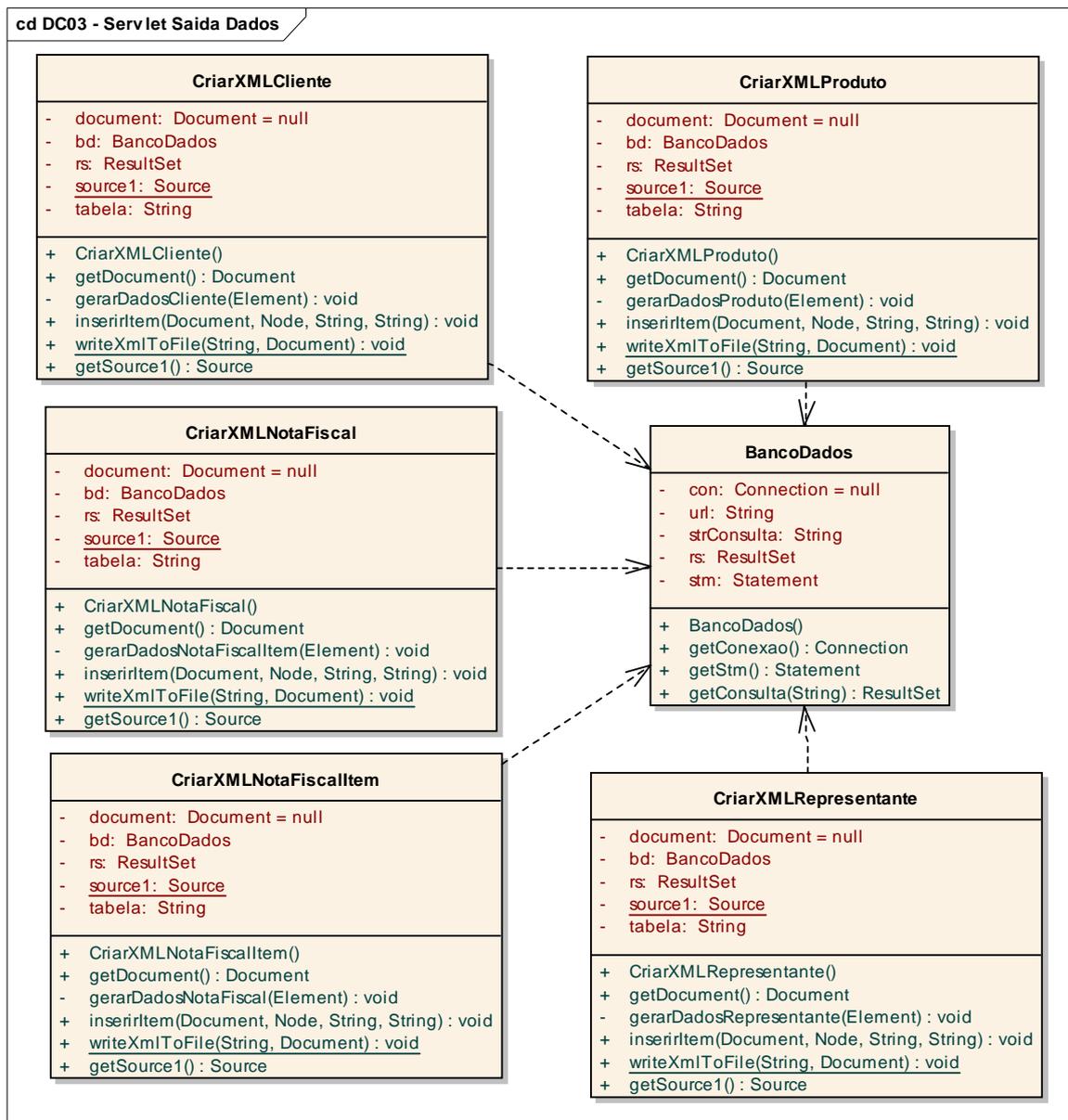


FIGURA 18 – Diagrama de classes de saída de dados

A figura 19 mostra as classes que tratam o recebimento dos dados do dispositivo móvel para o servidor.

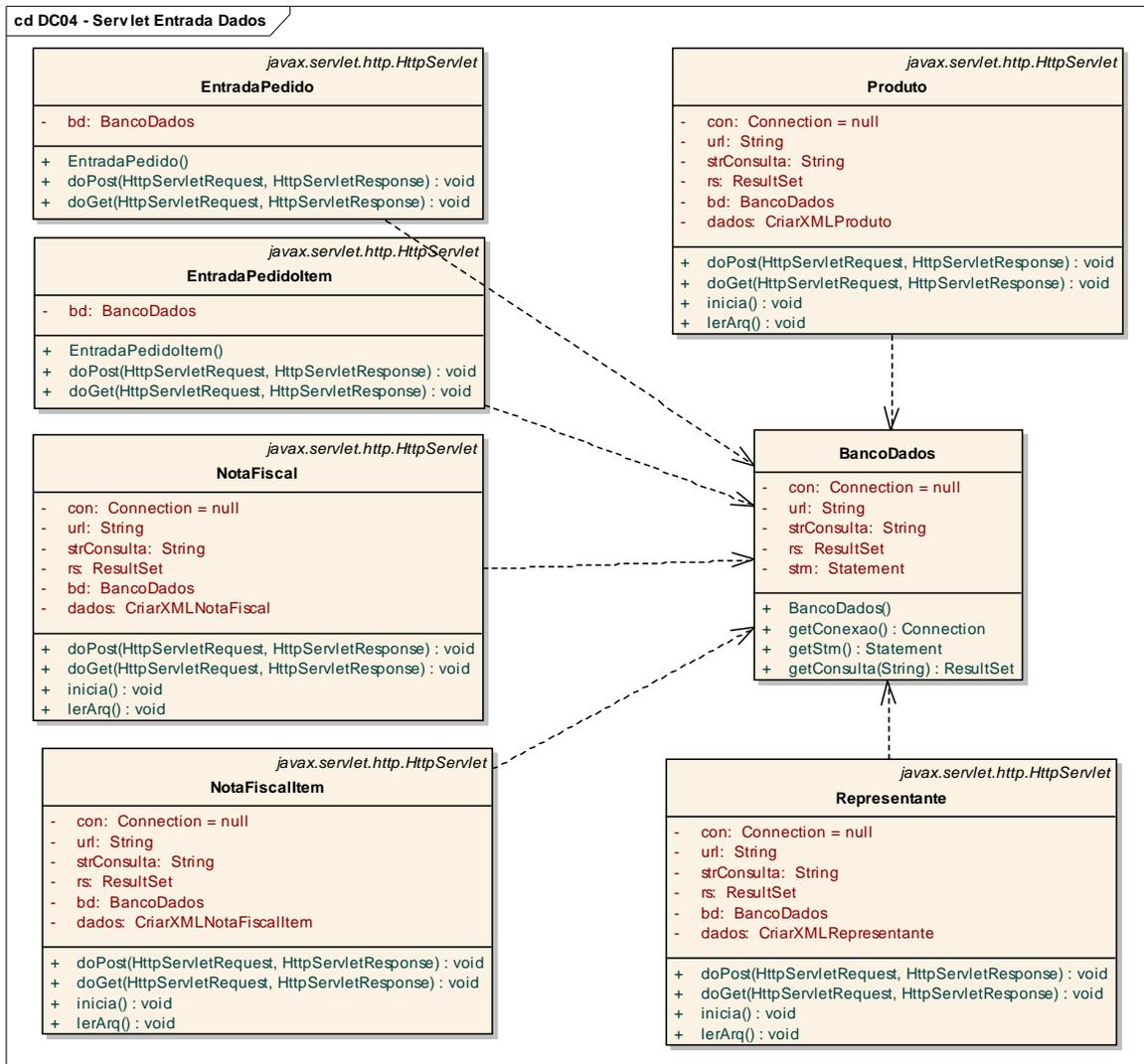


FIGURA 19 – Diagrama de classes de entrada de dados

### 3.3.5 MODELO ENTIDADE RELACIONAMENTO

O modelo entidade relacionamento é usado para a modelagem da base de dados, suas entidades e atributos e como estão relacionadas entre si.

A seguir são apresentados os modelos entidade relacionamento do banco de dados do

dispositivo móvel e do banco de dados da empresa.

A figura 20 mostra o modelo entidade relacionamento do banco de dados do dispositivo móvel.

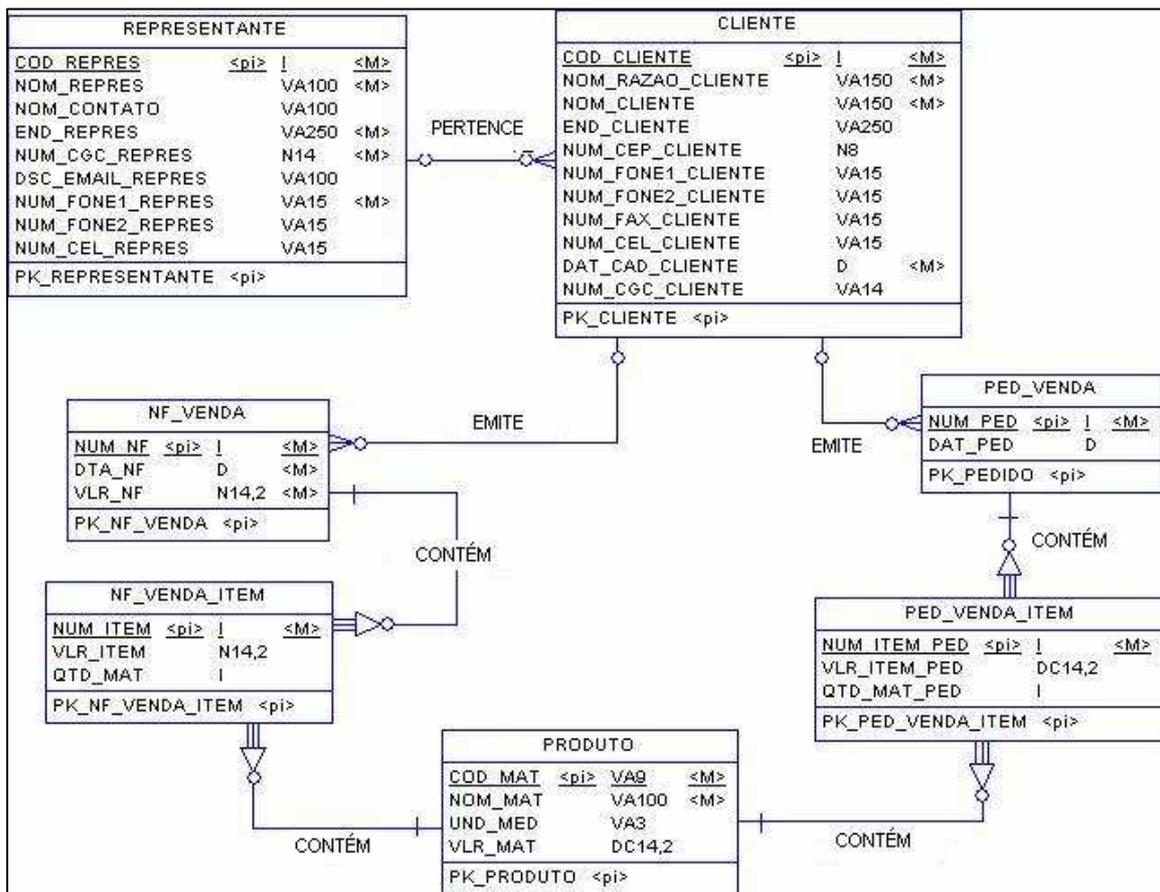


FIGURA 20 – Modelo entidade relacionamento da base de dados do dispositivo móvel

A figura 21 mostra o modelo entidade relacionamento do banco de dados do servidor da empresa.

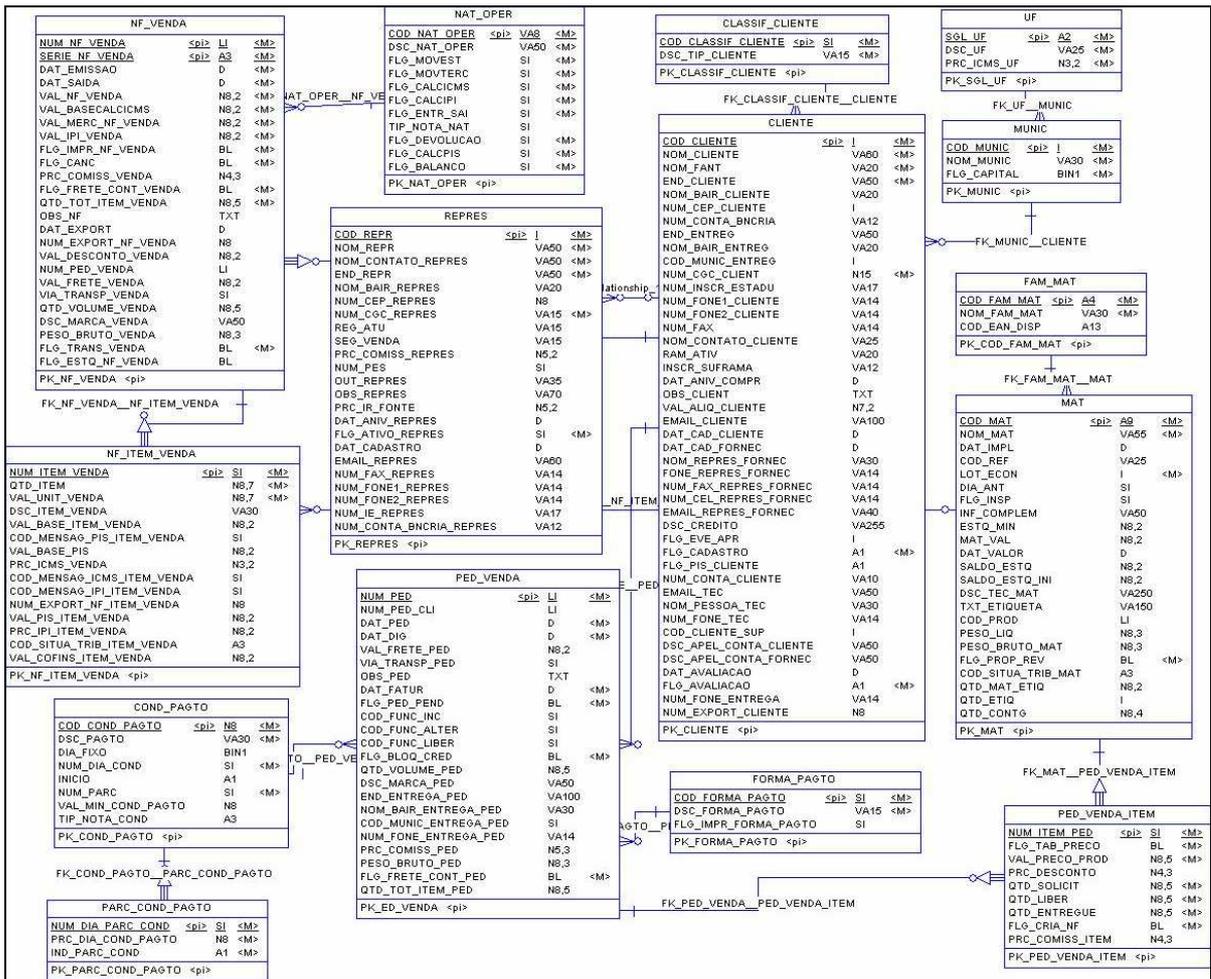


FIGURA 21 – Modelo entidade relacionamento da base de dados do servidor

### 3.4 IMPLEMENTAÇÃO

Os assuntos seguintes descrevem as ferramentas e técnicas utilizadas para o desenvolvimento do trabalho.

#### 3.4.1 TÉCNICAS E FERRAMENTAS UTILIZADAS

A seguir são descritas as funcionalidades de cada ferramenta utilizada no desenvolvimento do trabalho.

#### 3.4.1.1 Plataforma Eclipse

Eclipse é um ambiente de desenvolvimento integrado e extensível que possibilitou a codificação Java do sistema. Controle de projetos, compilação, depuração e execução de programas Java são alguns de seus recursos. Oferece também recursos de ajuda para montar projetos com todas as restrições necessárias a um projeto Java.

A característica extensível da ferramenta possibilitou a instalação de plugins que auxiliaram em atividades específicas dentro do desenvolvimento do trabalho. Foram utilizados dois *plugins* no desenvolvimento deste trabalho, o EclipseME e o Tomcat.

O *plugin* EclipseME, que foi acoplado à ferramenta Eclipse forneceu recursos para o desenvolvimento de aplicações Java para a *Micro Edition*. Permitiu a criação de projetos MIDlets, com toda a estrutura necessária para a implantação da aplicação. Possibilitou configurações especiais da ferramenta Eclipse que automatizaram a compatibilidade das funções APIs e utilização da máquina virtual do J2ME.

O Tomcat foi utilizado para a criação e execução de projetos *Servlets*. Este trabalho usufruiu da técnica de aplicações web para a comunicação do dispositivo com o banco de dados da empresa. Foi a partir do Tomcat que o que foi feita a troca de dados entre dispositivo e banco de dados da empresa.

#### 3.4.1.2 J2ME Wireless Toolkit

J2ME Wireless Toolkit é o ambiente de desenvolvimento de aplicações J2ME distribuído pela Sun Microsystems. Juntamente ao toolkit acompanha a máquina virtual KVM e emulador de dispositivos móveis. Estes recursos foram utilizados para o desenvolvimento dos aplicativos J2ME. As APIs do J2ME referenciadas no desenvolvimento também foram

disponibilizadas pelo toolkit.

### 3.4.1.3 Palm OS Simulator

Palm OS Simulator é um simulador do sistema operacional Palm OS.

Sua principal característica é que ele não emula o hardware, mas executa o mesmo sistema operacional encontrado no dispositivo Palm sobre uma camada de abstração de hardware. Com o simulador foi possível criar um ambiente idêntico ao que seria utilizado no dispositivo real, com configurações de memória, visor e conectividade.

### 3.4.2 Codificação

Foram criadas várias classes Java para o trabalho. A seguir são mostrados alguns trechos de código fazendo referências às principais características da aplicação, como conexão com banco de dados, conexão com servidor para troca de dados e montagem de telas para o usuário.

O quadro 2 mostra uma codificação utilizada para que a aplicação faça acesso ao banco de dados do *palmtop*.

```
/**
 * Faz a conexão com o banco de dados no Palm */
public void conecta(){
    try {
        // Carrega driver do DB2
        Class.forName("com.ibm.db2e.jdbc.DB2eDriver");
        // Faz conexao com banco TCCJ2ME
        con = DriverManager.getConnection("jdbc:db2e:TCCJ2ME");
        st = con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,
            ResultSet.CONCUR_UPDATABLE);
    }catch (Exception ex) {
        ex.printStackTrace();
    }
}
```

QUADRO 2 - Codificação do acesso ao banco de dados

O quadro 3 mostra uma codificação de consulta ao banco de dados e que retorna dados do representante para serem exibidos ao usuário.

```

try {
    stm = Principal.getCon().createStatement(
        ResultSet.TYPE_SCROLL_INSENSITIVE,
        ResultSet.CONCUR_UPDATABLE);
    //Executa a consulta no banco de dados
    rs = stm.executeQuery("SELECT COD_REPRES, NOM_REPRES, " +
        " END_REPRES, NOM_CONTATO, NUM_CGC_REPRES, " +
        "DSC_EMAIL_REPRES, NUM_FONE1_REPRES, " +
        "NUM_FONE2_REPRES, NUM_CEL_REPRES " +
        "FROM REPRESENTANTE");
    //enquanto houver dados no ResultSet apresenta para o usuário
    while(rs.next()){
        txtNomeRepres.setText(rs.getString("NOM_REPRES"));
        txtCNPJ.setString(rs.getString("NUM_CGC_REPRES"));
        txtEndereco.setString(rs.getString("END_REPRES"));
        txtEmail.setString(rs.getString("DSC_EMAIL_REPRES"));
        txtNomeContato.setString(rs.getString("NOM_CONTATO"));
        txtFone1.setString(rs.getString("NUM_FONE1_REPRES"));
        txtFone2.setString(rs.getString("NUM_FONE2_REPRES"));
        txtCelular.setString(rs.getString("NUM_CEL_REPRES"));
    }
} catch(SQLException ex){
    ex.printStackTrace();
}

```

QUADRO 3 - Codificação de consulta ao banco de dados

O quadro 4 traz uma codificação utilizada para fazer atualizações de dados na base de dados do dispositivo móvel.

```

try{
    //Executa o update na tabela REPRESENTANTE
    stm.executeUpdate("UPDATE REPRESENTANTE SET " +
        "END_REPRES = '" + txtEndereco.getString() + "' ," +
        "DSC_EMAIL_REPRES = '" + txtEmail.getString() + "' ," +
        "NUM_FONE1_REPRES = '" + txtFone1.getString() + "' ," +
        "NUM_FONE2_REPRES = '" + txtFone2.getString() + "' ," +
        "NUM_CEL_REPRES = '" + txtCelular.getString() + "' ," +
        "NUM_CGC_REPRES = '" + txtCNPJ.getString() + "' ," +
        "NOM_CONTATO = '" + txtNomeContato.getString() + "'");
    //Mostra mensagem com dados gravados
    alErro = new Alert("Mensagem", "Dados gravados !", null, null);
    alErro.setTimeout(Alert.FOREVER);
    Principal.getDisplay().setCurrent(alErro, this);
} catch(SQLException ex){
    ex.printStackTrace();
    alErro = new Alert("Erro", ex.getMessage() + data, null, null);
    alErro.setTimeout(Alert.FOREVER);
    Principal.getDisplay().setCurrent(alErro, this);
}

```

QUADRO 4 - Codificação de atualização de dados

O quadro 5 traz a codificação utilizada para fazer o acesso ao servidor da empresa e receber uma resposta com dados do arquivo XML.

```

public void run() {
    //Endereço servlet
    String url = "http://192.168.200.1/examples/servlet/serv.Dados" ;
    HttpURLConnection hc = null;
    InputStream in = null; //variavel que recebe os dados do servlet
    try {
        hc = (HttpURLConnection)Connector.open(url);
        in = hc.openInputStream();
        parse(in); //passa para o parser XML o que recebeu
    }
    catch (Exception e) {
        e.printStackTrace();
    }
    finally {
        if (in != null) {
            try { in.close(); }
            catch (IOException ioe) {}
        }
        if (hc != null) {
            try { hc.close(); }
            catch (IOException ioe) {}
        }
    }
}
}

```

QUADRO 5 - Codificação de conexão e recebimento de dados

O quadro 6 mostra o resumo de codificação de uma tela para o usuário.

```

public class FrmRepresentante extends Form implements CommandListener{

    private Command cmdVoltar; // Botões da tela
    private Command cmdSalvar;
    private Displayable frmAnterior; // Forms
    private StringItem txtNomeRepres; // Componentes da tela
    private TextField txtEndereco;

    public FrmRepresentante(Displayable d) {
        //Monta tela
        super("Dados do Representante");
        frmAnterior = d;
        txtNomeRepres = new StringItem("Representante:", "");
        txtEndereco = new TextField("Endereço:", "", 100, TextField.ANY);
        append(txtNomeRepres);
        append(txtEndereco);
        cmdVoltar = new Command("Voltar", Command.BACK,1);
        cmdSalvar = new Command("Gravar", Command.ITEM,1);
        addCommand(cmdVoltar);
        addCommand(cmdSalvar);
        setCommandListener(this);
    }
}

```

QUADRO 6 - Codificação de componentes de interface gráfica

### 3.4.3 Operacionalidade da implementação

Para demonstrar a operacionalidade da aplicação, são apresentados a seguir funções do sistema visando demonstrar um caso real da utilização do mesmo.

Ao carregar a aplicação, é disponibilizado para o usuário o menu principal do sistema, conforme a figura 22. Estão disponíveis 3 opções para o usuário. O primeiro sai da aplicação, o segundo mostra uma listagem dos clientes cadastrados e o terceiro abre o menu com todas as opções da aplicação.

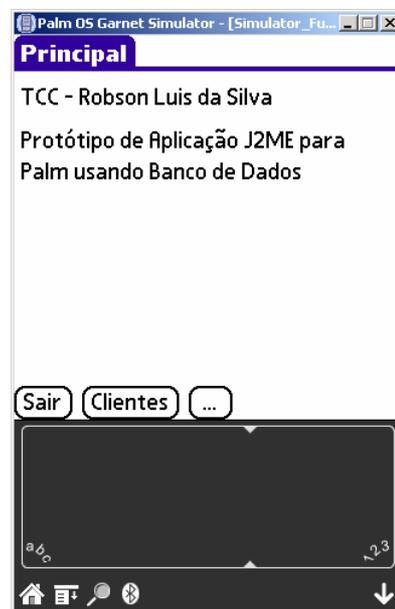


FIGURA 22 – Tela inicial

Ao selecionar a opção do menu, a aplicação mostra um menu para o usuário, conforme a figura 23, as opções disponíveis para o usuário são:

- a) clientes;
- b) dados do representante;
- c) produtos;
- d) faturamento;
- e) importar dados;

- f) exportar dados;
- g) limpar base de dados;
- h) sair.



FIGURA 23 – Menu de opções

Selecionando clientes, o programa mostra uma tela para o usuário com os dados dos clientes cadastrados no banco de dados. Esta tela está representada na figura 24.

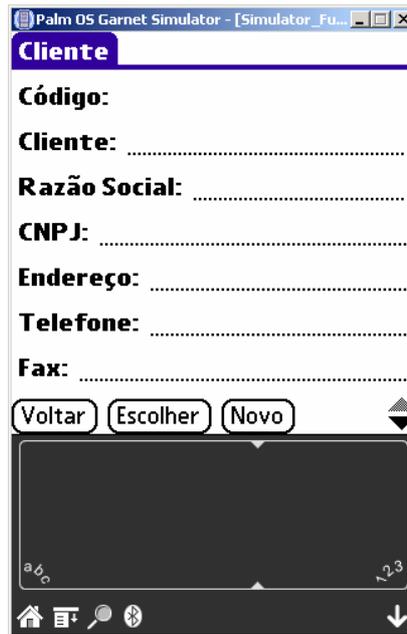


FIGURA 24 – Tela de clientes

O sistema apresenta nesta tela mais três opções para o usuário: voltar para o menu principal, escolher um cliente para ver os seus dados ou cadastrar um novo cliente.

Ao selecionar a opção de escolha o sistema mostra a tela representada na figura 25, relacionando os clientes já cadastrados.

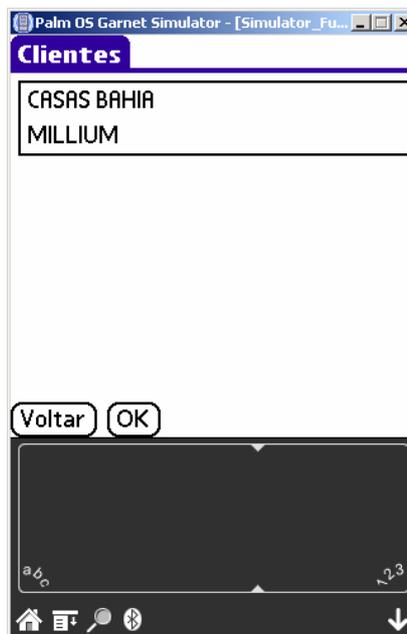


FIGURA 25 – Seleção de clientes cadastrados

Ao selecionar um cliente e confirmar, o sistema mostra a tela do cadastro de clientes com os dados do cliente selecionado, como mostra a figura 26. Neste momento o botão para incluir um novo cliente é substituído pelo botão de gravação de dados, para que o usuário possa salvar as alterações feitas no cadastro do cliente.



FIGURA 26 – Dados do cliente selecionado

Conforme mostrado na figura 26, a tela de clientes tem a opção de cadastramento de novos cliente. Ao selecionar esta opção o sistema sugere um código de cliente novo e mostra a tela para que o usuário faça a entrada dos dados, conforme mostra a figura 27.



Palm OS Garnet Simulator - [Simulator\_Fu...]

**Cliente**

**Código:** 3

**Cliente:** .....

**Razão Social:** .....

**CNPJ:** .....

**Endereço:** .....

**Telefone:** .....

**Fax:** .....

Voltar Escolher Gravar

a6 7,3

FIGURA 27 – Cadastramento de novo cliente

A segunda opção no menu principal do sistema, possibilita o usuário ver as informações do representante que tem os dados carregados no dispositivo móvel, conforme ilustrado na figura 28. Os dados do representante são mantidos pela base de dados da empresa e são carregados para o dispositivo móvel quando o usuário faz uma importação de dados.

Nesta tela é possível somente alterar os dados que estão cadastrados previamente, não sendo possível o cadastramento de novos representantes.

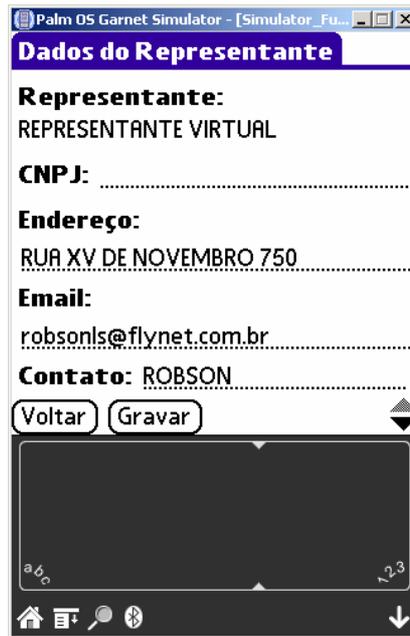


FIGURA 28 – Dados do representante

A terceira opção no menu principal do sistema é a listagem de produtos. Esta lista de produtos não pode ser alterada, pois os dados vêm diretamente da empresa ao atualizar os dados do sistema no dispositivo móvel. A tela é mostrada na figura 29.

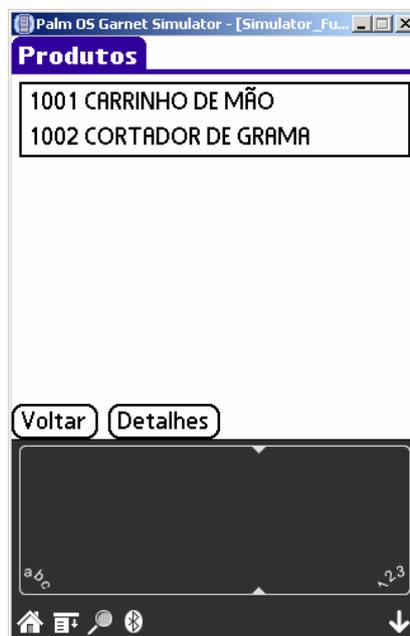


FIGURA 29 - Lista de produtos

Nesta tela é possível o usuário ver os detalhes do produto selecionado, conforme a figura 30.

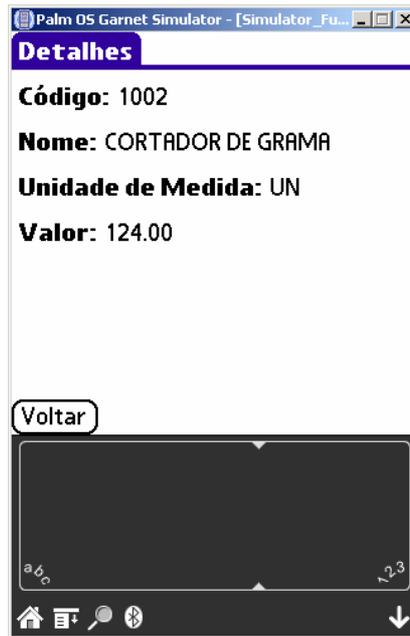


FIGURA 30 – Detalhes do produto

No menu principal o usuário tem a opção pedidos, onde o sistema exibirá a tela conforme a figura 31, que possibilita ao usuário consultar os pedidos que estão armazenados no banco de dados. Esta tela também tem a opção de cadastramento de novos pedidos.

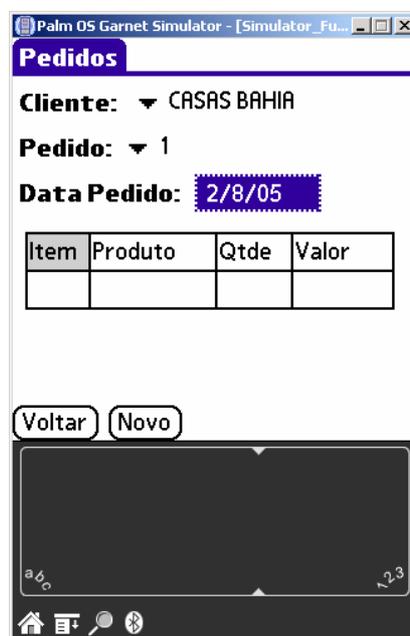


FIGURA 31 – Tela de pedidos

Ao selecionar novo pedido o sistema mostrará a tela sugerindo um novo número de pedido. O usuário poderá escolher o cliente e incluir itens no pedido, conforme mostra a

figura 32.

Palm OS Garnet Simulator - [Simulator\_Fu...]

**Pedidos**

Cliente: ▼ CASAS BAHIA

Pedido: 7.....

Data Pedido: 3/2/05

Item	Produto	Qtde	Valor

Novo Item Gravar ...

abc 123

Home Back Search Bluetooth

FIGURA 32 – Inclusão de pedido

Selecionando novo item o sistema abrirá a tela para digitação de itens do pedido, conforme mostra a figura 33.

Palm OS Garnet Simulator - [Simulator\_Fu...]

**Item do Pedido**

Pedido: 7.....

Item: 1.....

Código: ▼ 1001

Descrição: ▼ CARRINHO DE MÃO

Unidade Medida: UN.....

Valor: 23.00.....

Quantidade: 2.....

Voltar Gravar

abc 123

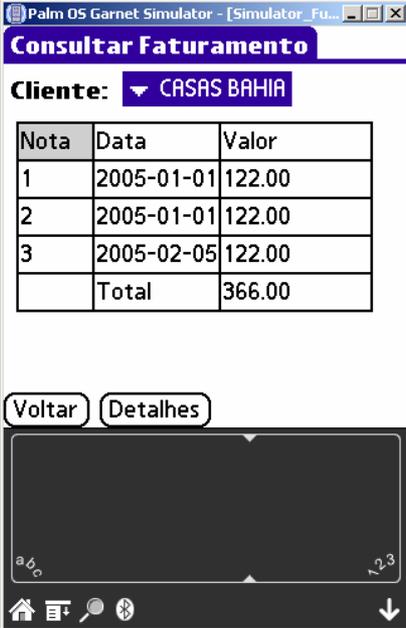
Home Back Search Bluetooth

FIGURA 33 – Inclusão de itens do pedido

Gravando o item o sistema voltará para a tela do pedido.

No menu principal o usuário tem a opção de consultar o faturamento dos seus clientes

através da opção faturamento. O sistema exibe uma tela com as notas fiscais faturadas do cliente selecionado, como mostra a figura 34. Estes dados não podem ser alterados no dispositivo móvel, pois eles provêm do servidor da empresa e estão disponíveis apenas para consultas.



Nota	Data	Valor
1	2005-01-01	122.00
2	2005-01-01	122.00
3	2005-02-05	122.00
	Total	366.00

FIGURA 34 – Consulta faturamento

Esta tela possibilita ao usuário uma consulta aos itens que foram faturados na nota fiscal selecionada. Selecionando o botão detalhes, o sistema abre uma nova tela com os detalhes conforme visto na figura 35.

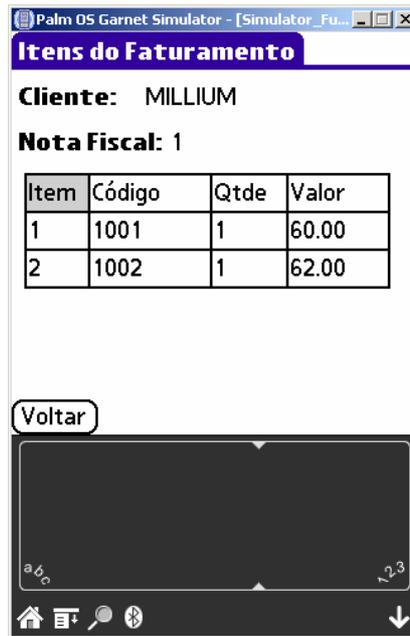


FIGURA 35 – Detalhes da nota fiscal

Ao selecionar a opção importar dados do menu principal o sistema efetua uma conexão com o servidor da empresa, que possui o *servlet java* pronto para atender as requisições feitas, atualiza os dados que estão no banco de dados do *palmtop* e exibe uma mensagem para o usuário, conforme a figura 36.

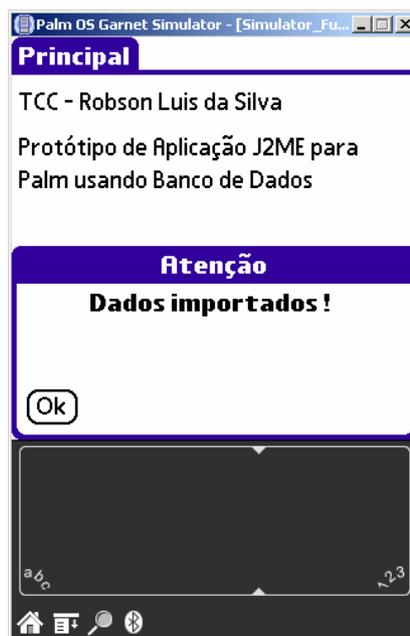


FIGURA 36 – Importação de dados

A opção exportar dados é semelhante a opção de importação, porém o sistema faz uma conexão com o servidor e envia os dados a serem cadastrados no servidor da empresa. Esta conexão é feita através do protocolo HTTP que faz o acesso ao servlet que trata a requisição. Os dados a serem gravados na base de dados da empresa são enviados juntos com a requisição, o *servlet* responsável trata esses parâmetros e grava os dados na base de dados da empresa. Após a exportação dos dados, a aplicação exibe uma mensagem conforme a figura 37.

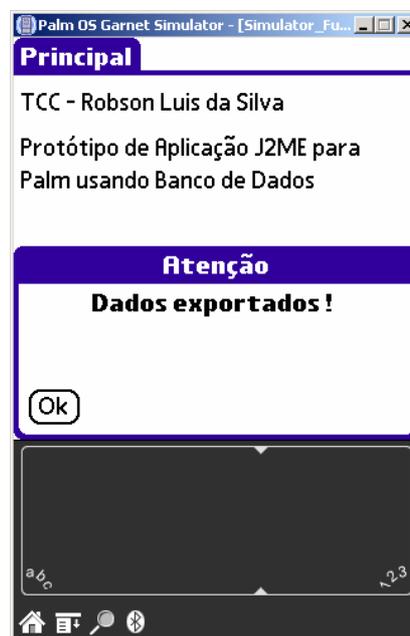


FIGURA 37 – Exportação de dados

A opção de limpeza de base apaga todas as tabelas do banco de dados do dispositivo móvel e as cria novamente. Após isso o sistema emite uma mensagem para o usuário informando que a base de dados não possui mais dados, conforme figura 38.

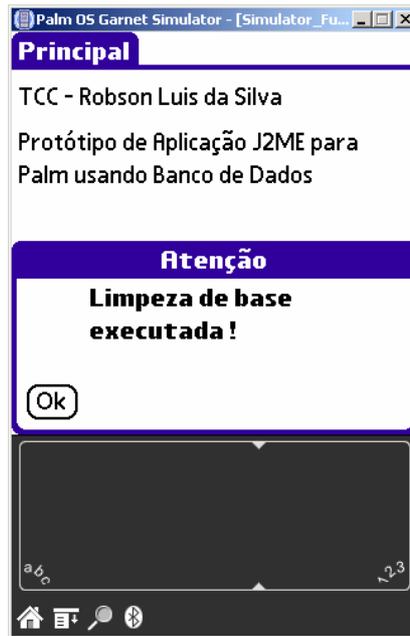


FIGURA 38 – Limpeza da base de dados

### 3.5 RESULTADOS E DISCUSSÃO

Os testes do protótipo foram feitos no simulador e também em um dispositivo móvel real. Os casos de uso foram testados e foi verificado que o sistema atendeu aos requisitos especificados.

O dispositivo real utilizado para os testes foi o Palm Tungsten E, cujas características são processador de 126 Mhz, 32 Mb de memória e sistema operacional PalmOS versão 5.2.1.

No protótipo é possível fazer uma conexão ao servidor da empresa que mantém os dados dos representantes e realizar atualizações na base de dados que fica no dispositivo móvel. A importação dos dados permite ao usuário consultar o faturamento dos seus clientes, assim como usar os clientes e produtos para relacioná-los e incluir novos pedidos que são enviados para a empresa. Os dados recebidos do servidor da empresa e os dados que são

inseridos pelo usuário são gravados em um banco de dados relacional instalado no *palmtop*.

O uso do banco de dados IBM DB2 Everyplace mostrou-se viável, auxiliando na recuperação e na gravação dos dados na base de dados do dispositivo móvel. O uso de JDBC pode ser feito perfeitamente através da API disponibilizada pelo fornecedor.

Nos testes realizados no software o banco de dados recebeu uma carga de dados relativamente pequena em relação à base de dados da empresa, com isso não houve nenhum problema em relação à *performance* na recuperação e gravação dos dados.

Em Schmitt Junior (2004), o trabalho faz uma conexão com um servidor na empresa, usando um *servlet java*, efetuando a troca de dados entre o dispositivo móvel, que no caso é um celular e o servidor, porém a persistência de dados utilizada naquele trabalho não é em um banco de dados relacional. Diferentemente do que foi citado por ele, o uso de JDBC no dispositivo móvel mostrou-se perfeitamente possível no protótipo desenvolvido.

Nesta troca de dados, o dispositivo recebe do servidor um arquivo XML.

Em Kracik (2002), é demonstrado o uso de arquivos XML em um sistema que atua em um computador desktop. No protótipo apresentado naquele trabalho o uso de arquivos XML se dá também em dispositivos móveis, que possuem uma certa limitação em relação ao desktop.

O protótipo desenvolvido neste trabalho utiliza configurações e perfis diferentes dos usados em outros trabalhos. Um exemplo é o trabalho de Depiné (2002), onde é desenvolvido um protótipo de software para cálculo de uma planilha de rally de regularidade, que foi desenvolvido utilizando-se o perfil MIDP 1 e a configuração CLDC 1.0. Neste trabalho o protótipo foi desenvolvido utilizando-se o perfil MIDP 2 e configuração CLDC 1.1, onde a maior diferença são o componente *CustomItem* que permite o desenvolvedor criar componentes gráficos customizados e a inclusão do suporte a cálculos com ponto flutuante.

O desenvolvimento deste trabalho está voltado para a plataforma *Palm*, onde pode ser

utilizado um banco de dados relacional para armazenamento de dados. Em Galvin (2004), é apresentado um protótipo de sistema para dispositivos móveis visando auxiliar o trabalho de relacionamento entre empresas e clientes utilizando dispositivos móveis do tipo *Pocket PC*, o sistema operacional *Windows .NET CE* e o banco de dados *Microsoft SQL Server 2000 CE*.

Como citado, todas as conexões feitas pelo protótipo ao servidor da empresa, são feitas através do protocolo *Hipertext Transfer Protocol* (HTTP) e fazem o acesso a *servlets java*. Esta foi uma alternativa diferente da encontrada no trabalho de Schaefer (2004), onde o sistema desenvolvido para envio de mensagens para uma empresa de transporte, utilizando o protocolo *Simple Mail Transfer Protocol* (SMTP).

### 3.5.1 PROBLEMAS ENCONTRADOS DURANTE O DESENVOLVIMENTO

Na proposta original do trabalho, era previsto que o usuário teria os arquivos para importação de dados da empresa localizados no dispositivo móvel, assim a implementação carregaria um arquivo XML diretamente do dispositivo móvel, porém a configuração CLDC não tem suporte a protocolos que trabalhem com sistemas de arquivo. Sendo assim, a solução encontrada foi fazer com que o dispositivo conectasse a um servidor através de protocolo HTTP e recebesse as respostas em arquivos do formato XML.

Outra dificuldade encontrada foi em relação a parte gráfica, pois apesar de a MIDP 2 implementar componentes novos e melhorar muitos já existentes na MIDP 1, não existe um componente em forma de grid para apresentação de dados. A solução encontrada foi implementar um componente customizado utilizando a classe *CustomItem* presente na MIDP 2, que por sua vez permite que o usuário crie o seu próprio componente, desenhando através de *Canvas*.

## 4 CONCLUSÕES

O protótipo desenvolvido cumpriu os requisitos especificados, permitindo a troca de dados entre servidor e dispositivo móvel. O sistema permite ao usuário acesso a dados da empresa que antes não eram possíveis, como o faturamento da sua carteira de clientes.

O sistema possibilita também ao usuário fazer a entrada de dados dos pedidos que devem ser enviados para a empresa, trabalho que é feito manualmente sem o uso de dispositivos móveis.

A portabilidade do sistema, algo muito difundido através da plataforma Java, não fica comprometida neste trabalho, visto que o uso de banco de dados relacional, neste caso o *IBM DB2 Everyplace*, está disponível para a maioria das plataformas onde se pode utilizar J2ME, como por exemplo, sistemas operacionais *Symbian*, *Windows CE*, entre outros.

O uso de um banco de dados relacional no dispositivo móvel mostrou-se perfeitamente viável, auxiliando na recuperação de dados, visto que o desenvolvedor pode fazer uso da *Structured Query Language (SQL)*.

As ferramentas usadas para o desenvolvimento atenderam aos requisitos necessários para a implementação, pois a máquina virtual Java e o banco de dados relacional estavam disponíveis tanto para o simulador Palm, quanto para o dispositivo móvel usado. A linguagem Java estava disponível no J2ME Wireless Toolkit e pôde ser utilizada perfeitamente para a construção do aplicativo.

Os testes do protótipo foram feitos em um ambiente de simulação utilizando o Palm Os Simulator e um dispositivo móvel real.

Um dos maiores objetivos deste trabalho a ser alcançado, era adquirir o conhecimento sobre a plataforma J2ME visando o uso desta tecnologia no ambiente de trabalho, e este foi atingido.

#### 4.1 EXTENSÕES

Como extensões deste trabalho podem ser sugeridas algumas opções.

A portabilidade deste protótipo para outras plataformas como Windows CE, ou até mesmo verificar a viabilidade deste sistema em um celular.

O desenvolvimento deste trabalho utilizando outro banco de dados relacional diferente do que foi utilizado.

O desenvolvimento deste trabalho utilizando outra configuração J2ME, como CDC, onde o uso de sistemas de arquivos é possível.

A continuação deste trabalho visando um maior controle dos dados que são importados e exportados e também um controle sobre os dados que são alterados no dispositivo móvel.

## REFERÊNCIAS BIBLIOGRÁFICAS

- CORBERA, Rodrigo Garcia. **Tutorial de programação J2ME**. [S.l.],[2003]. Disponível em: <[http://geocities.yahoo.com.br/brasilwireless/tutorial\\_j2me/j2me\\_01/j2me\\_01.html](http://geocities.yahoo.com.br/brasilwireless/tutorial_j2me/j2me_01/j2me_01.html)>. Acesso em: 4 out. 2005.
- DEITEL, Harvey M.; DEITEL, Paul J.; NIETO, Tem; LIN, Ted; SADHU, Praveen. **XML: how to program**. Maynard. Prentice Hall, 2001.
- DELL INC. **Dell home systems Axim X30**. [S.l.], 2005. Disponível em: <[http://www1.us.dell.com/content/products/productdetails.aspx/axim\\_x30\\_high?c=us&cs=19&l=en&s=dhs](http://www1.us.dell.com/content/products/productdetails.aspx/axim_x30_high?c=us&cs=19&l=en&s=dhs)>. Acesso em: 06 abr. 2005.
- DEPINÉ, Fábio Marcelo. **Protótipo de software para dispositivos móveis utilizando Java ME para cálculo de regularidade em rally**. 2002. 55 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- FONSECA, Jorge Cavalcanti. **Portando a KVM**. 2002. 64 f. Trabalho de Conclusão de Curso (Graduação em Ciências da Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife.
- FURGERI, Sérgio. **Ensino didático da linguagem XML: aprenda a criar padrões e documentos inteligentes com a XML**. São Paulo: Érica, 2001.
- GALVIN, Deleon. **Protótipo de sistema CRM para dispositivos móveis utilizando a tecnologia .NET**. 2004. 90 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- GOYA, M. **A Família DB2**. São Paulo, [2004]. Disponível em: <<http://www.imasters.com.br/artigo.php?cn=2215&cc=60>> Acesso em: 13 out. 2005.
- IBM. **WebSphere everyplace micro environment**. [S.1], 2005. Disponível em <<http://www-306.ibm.com/software/wireless/weme>>. Acesso em: 8 jun. 2005.
- KNUDSEN, Jonathan. **Wireless Java: developing with J2ME**. 2 ed. Berkeley. Apress, 2003.
- KRACIK, João. **Protótipo de padrão de comunicação e armazenamento de dados contábeis usando XML**. Furb, 2002. 109 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- MATEUS, Geraldo R. **Introdução a computação móvel**. Rio de Janeiro: DCC/IM; COPPE Sistemas, 1998. 189 p.

MUCHOW, John W. **Core J2ME technology & MIDP**. Palo Alto. Prentice Hall. 1. ed. 2001.

PALMBRASIL. **Conheça o Palm**. [S.1], 2005. Disponível em: <<http://www.palmbrasil.com.br/conheca-palm/index.html>> Acesso em: 6 nov. 2005.

PALMONE. **Tungsten T5 Handled**. [S.1.], 2005. Disponível em: <<http://www.palmone.com/us/products/handhelds/tungsten-t5/specs.epl>>. Acesso em: 06 abr. 2005.

PALMSOURCE. **Palm OS**. [S.1], 2005. Disponível em: <<http://www.palmsource.com/palmos/>> Acesso em: 6 nov. 2005.

RODHES, Neil.;MCKEEHAN, Julie. **Palm programing: the developer's guide**. O'Reilly. 1. ed. 2001.

SCHAEFER, Carine. **Protótipo de aplicativo para transmissão de dados a partir de dispositivos móveis aplicado a uma empresa de transporte**. 2004. 53 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

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

SUN. **Java technology**. [S.1.], 2004. Disponível em: <<http://java.sun.com/>>. Acesso em: 12 mai. 2005.

WILDING-MCBRIDE, Daryl. **Java development on PDAs: building applications for PocketPC and Palm devices**. Boston. Addison-Wesley. 1. ed. 2003.

## **APÊNDICE A – Detalhamento dos casos de uso do sistema**

### **Caso de uso: cadastrar cliente**

O representante, ao entrar na aplicação tem no menu uma opção para cadastro de clientes. Uma nova tela é mostrada para o usuário com os seguintes dados a serem cadastrados:

- a) Código;
- b) razão social;
- c) nome de fantasia;
- d) CNPJ do cliente;
- e) telefone:
- f) celular;
- g) fax;
- h) endereço.

Antes de persistir os dados no banco de dados a aplicação irá verificar a validade do CNPJ cadastrado pelo representante.

Ao cadastrar um cliente novo na base de dados do dispositivo móvel, a aplicação sugere um código de cliente válido.

### **Caso de uso: dados do representante**

O sistema possibilita ao usuário consultar as informações do representante que tem os dados carregados no dispositivo móvel. Ao selecionar esta opção o usuário recebe uma tela mostrando as seguintes informações:

- a) nome do representante;
- b) CNPJ do representante;

- c) endereço;
- d) email;
- e) nome da pessoa de contato;
- f) telefone;
- g) fax;
- h) telefone celular.

### **Caso de uso: cadastrar pedido**

O representante, na tela principal da aplicação, tem no menu a opção para cadastro de pedidos que ao ser selecionada leva a uma nova tela com as seguintes informações a serem cadastradas.

- a) número do pedido;
- b) cliente;
- c) data de faturamento.

A aplicação irá fornecer para o usuário uma tela para a escolha dos clientes, que já estão cadastradas no banco de dados e provêm do sistema da empresa quando é feita a importação dos dados.

Após o cadastro destes dados, que são considerados o cabeçalho do pedido, a aplicação irá para outra tela para que o representante cadastre os produtos que são relacionados com esse pedido. Os dados a serem cadastrados pelo representante são:

- a) código do produto;
- b) quantidade do produto;
- c) valor unitário do produto;
- d) valor total do produto.

O código do produto será selecionado em uma tela seguinte com o cadastro dos

produtos da empresa. O valor total do produto será calculado pela aplicação.

### **Caso de uso: listar produtos**

O representante tem a opção de listar todos os produtos que estão na base de dados do dispositivo móvel. Os produtos são cadastrados na base de dados do dispositivo móvel ao realizar a importação dos dados. Ao selecionar a opção para listar os produtos, a aplicação mostrará uma tela com a lista com o código e o nome de todos os produtos que estão na sua base de dados. O representante pode selecionar um produto e visualizar os detalhes deste produto.

### **Caso de uso: listar clientes**

O representante tem a opção de listar todos os clientes que estão na sua base de dados no dispositivo móvel. A aplicação irá mostrar uma tela com os dados do cliente selecionado, nesta tela o usuário terá uma opção para ver todos os clientes cadastrados, esta opção irá listar o nome de cada cliente em uma tela específica. O usuário poderá nesta lista escolher o cliente que desejar e então o sistema voltará para a tela anterior com os dados do cliente.

### **Caso de uso: listar faturamento**

O usuário, na tela principal da aplicação, tem a opção de listar faturamento. Ao selecionar essa opção o programa pedirá um cliente e listará todo o faturamento deste cliente.

As seguintes informações serão mostradas para o usuário;

- a) data de faturamento;
- b) nota fiscal;
- c) valor total da nota fiscal.

Ao selecionar uma nota fiscal o usuário pode ver os detalhes dessa nota, tais como produto, valor unitário e valor total faturado.

**Caso de uso: listar pedido**

O usuário, na tela principal da aplicação, tem a opção de listar pedidos cadastrados. Ao selecionar essa opção o programa pedirá um cliente e listará todos os pedidos cadastrados desse cliente.

As seguintes informações serão mostradas para o usuário:

- a) número do pedido;
- b) data do pedido.

Ao escolher um determinado pedido, a aplicação mostra a tela para cadastro de pedidos com os dados do respectivo pedido.

**Caso de uso: importar dados**

Existe um sistema hoje onde todos os dados já estão cadastrados. A aplicação importa todas as informações necessárias fazendo uma conexão com o servidor da empresa e recebendo os dados em um arquivo no formato XML gerado pelo sistema. As seguintes tabelas são importadas pela aplicação;

- a) clientes;
- b) notas fiscais;
- c) itens de notas fiscais;
- d) produtos;
- e) pedidos;
- f) itens de pedidos;
- g) representante.

Os dados importados são inseridos no banco de dados que está instalado no dispositivo móvel.

**Caso de uso: exportar dados**

A aplicação faz uma conexão com o servidor da empresa e envia os dados que irão gerar um arquivo no formato XML que é importado pelo sistema existente hoje na empresa.

São dados exportados no arquivo:

- a) clientes novos;
- b) pedidos novo.

**Caso de uso: limpar base de dados**

O sistema faz um acesso à base de dados do dispositivo móvel e recria todas as tabelas relacionadas.

## APÊNDICE B – Estrutura dos arquivos XML

Arquivo de cliente: o quadro 7 mostra a estrutura do arquivo Cliente.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Cliente (cod_cliente | cod_repres | cod_classif |
nom_razao_cliente | nom_fant_cliente | num_cep_cliente |
num_cel_cliente | num_fone1_cliente | num_fone2_cliente |
num_fax_cliente | dat_cad_cliente)+>
<!ELEMENT Dados (Cliente)>
<ELEMENT cod_classif (#PCDATA)>
<ELEMENT cod_cliente (#PCDATA)>
<ELEMENT cod_repres (#PCDATA)>
<ELEMENT dat_cad_cliente (#PCDATA)>
<ELEMENT nom_fant_cliente (#PCDATA)>
<ELEMENT nom_razao_cliente (#PCDATA)>
<ELEMENT num_cel_cliente (#PCDATA)>
<ELEMENT num_cep_cliente (#PCDATA)>
<ELEMENT num_fax_cliente (#PCDATA)>
<ELEMENT num_fone1_cliente (#PCDATA)>
<ELEMENT num_fone2_cliente (#PCDATA)>
```

QUADRO 7 – Estrutura do arquivo Cliente.dtd

Arquivo de representantes: o quadro 8 mostra a estrutura do arquivo Representante.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Dados (Representante)>
<!ELEMENT Representante (cod_repres, nom_repres, nom_contato,
end_repres, num_cep_repres, num_cgc_repres, num_fone1_repres,
num_fone2_repres, num_cel_repres, dsc_email_repres,
dat_cad_repres)>
<ELEMENT cod_repres (#PCDATA)>
<ELEMENT dat_cad_repres (#PCDATA)>
<ELEMENT dsc_email_repres (#PCDATA)>
<ELEMENT end_repres (#PCDATA)>
<ELEMENT nom_contato (#PCDATA)>
<ELEMENT nom_repres (#PCDATA)>
<ELEMENT num_cel_repres (#PCDATA)>
<ELEMENT num_cep_repres (#PCDATA)>
<ELEMENT num_cgc_repres (#PCDATA)>
<ELEMENT num_fone1_repres (#PCDATA)>
<ELEMENT num_fone2_repres (#PCDATA)>
```

QUADRO 8 – Estrutura do arquivo Representante.dtd

Arquivo de produtos: o quadro 9 mostra a estrutura do arquivo Produto.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Dados (Produto)>
<!ELEMENT Produto (cod_prod | nom_prod | und_med | vlr_mat)+>
<!ELEMENT cod_prod (#PCDATA)>
<!ELEMENT nom_prod (#PCDATA)>
<!ELEMENT und_med (#PCDATA)>
<!ELEMENT vlr_mat (#PCDATA)>
```

QUADRO 9 – Estrutura do arquivo Produto.dtd

Arquivo de notas fiscais: o quadro 10 mostra a estrutura do arquivo NotaFiscal.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Dados (NotaVenda)>
<!ELEMENT NotaVenda (num_nf | cod_cliente | dat_emissao |
val_nf)+>
<!ELEMENT cod_cliente (#PCDATA)>
<!ELEMENT dat_emissao (#PCDATA)>
<!ELEMENT num_nf (#PCDATA)>
<!ELEMENT val_nf (#PCDATA)>
```

QUADRO 10 – Estrutura do arquivo NotaFiscal.dtd

Arquivo de itens de nota fiscal: o quadro 11 mostra a estrutura do arquivo NotaFiscalItem.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Dados (NotaVenda)>
<!ELEMENT NotaVenda (num_nf | num_item | cod_prod | val_prod_item
| qtd_prod)+>
<!ELEMENT cod_prod (#PCDATA)>
<!ELEMENT num_item (#PCDATA)>
<!ELEMENT num_nf (#PCDATA)>
<!ELEMENT qtd_prod (#PCDATA)>
<!ELEMENT val_prod_item (#PCDATA)>
```

QUADRO 11 – Estrutura do arquivo NotaFiscalItem.dtd