

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO**  
(Bacharelado)

**APLICATIVO PARA AUXÍLIO NA EMISSÃO DOS AUTOS  
DE INFRAÇÕES DE TRÂNSITO NO MUNICÍPIO DE  
BLUMENAU**

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO À UNIVERSIDADE  
REGIONAL DE BLUMENAU PARA A OBTENÇÃO DOS CRÉDITOS NA  
DISCIPLINA COM NOME EQUIVALENTE NO CURSO DE CIÊNCIAS DA  
COMPUTAÇÃO — BACHARELADO

**ARQUELAU PASTA**

BLUMENAU, JUNHO/2003

2003/01-06

# **APLICATIVO PARA AUXÍLIO NA EMISSÃO DOS AUTOS DE INFRAÇÕES DE TRÂNSITO NO MUNICÍPIO DE BLUMENAU**

**ARQUELAU PASTA**

ESTE TRABALHO DE CONCLUSÃO DE CURSO FOI JULGADO ADEQUADO  
PARA OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA DE TRABALHO DE  
CONCLUSÃO DE CURSO OBRIGATÓRIA PARA OBTENÇÃO DO TÍTULO DE:

**BACHAREL EM CIÊNCIAS DA COMPUTAÇÃO**

---

Prof. Dr. Oscar Dalfovo — Orientador na FURB

---

Prof. José Roque Voltolini da Silva — Coordenador do  
TCC

**BANCA EXAMINADORA**

---

Prof. Dr. Oscar Dalfovo

---

Prof. Paulo de Tarso Mendes Luna, M.Eng.

---

Prof. Alexander Roberto Valdameri, M.Eng.

# **AGRADECIMENTOS**

Em primeiro lugar, a Deus, por tudo.

A minha esposa Silvana e minha filha Nadine pela compreensão, dedicação e minha ausência durante estes anos de estudo.

Aos meus pais, Orlando e Nair, pela compreensão, apoio e carinho durante estes anos de estudo.

Ao meu orientador e amigo, Prof. Dr. Oscar Dalfovo, pela orientação sempre oportuna, esclarecedora, inteligente e pelo seu incentivo, fazendo com que este trabalho pudesse ser realizado.

Aos amigos, aqueles que, direta ou indiretamente estiveram ao meu lado durante esta caminhada e contribuíram de alguma forma para a realização deste trabalho.

Em especial aos meus amigos Adriano Luiz Moretti e David Jones Treis, que em muito me ajudaram para a realização deste trabalho.

## RESUMO

Este trabalho apresenta um aplicativo para auxiliar no preenchimento do auto de infração de trânsito na cidade de Blumenau. O aplicativo foi desenvolvido para servir de substituto do bloco de papel, utilizado pelos agentes de trânsito, quando da emissão do auto de infração. Na seqüência este auto é encaminhado para um arquivo texto, que será transmitido para o órgão responsável pela emissão da notificação. Este aplicativo elimina o trabalho de digitação os autos de infrações e reduz o número de cancelamento de notificações. O aplicativo faz uso de um computador de mão, chamado *hand-held ou palmtop*, o que o torna fácil e leve para o manuseio.

## **ABSTRACT**

This work presents a applicatory one to assist in the fulfilling of the finding of infraction of transit in the city of Blumenau. The applicatory one was developed to serve of substitute of the block of paper, used for the agents of transit, when of the emission of the tax assessment notice. In the sequence this auto one is directed for an archive text, that will be transmitted for the responsible agency for the emission of the notification. This applicatory one eliminates the work of keyboarding of files of legal documents of infractions and reduces the number of cancellation of notifications. The applicatory one makes use of a hand computer, call hand-held or palmtop, what it becomes it easy and it has led for the manuscript.

## LISTA DE FIGURAS

|  |    |
|--|----|
| Figura 1 - Exemplo de um computador <i>hand-held</i> .....           | 17 |
| Figura 2 - Exemplo de um computador <i>Palmtop</i> .....             | 18 |
| Figura 3 - Impacto da alteração da realidade .....                   | 33 |
| Figura 4 - Ciclos desenho - protótipo e desenho – produção .....     | 34 |
| Figura 5 - Representação de um processo .....                        | 36 |
| Figura 6 - Representação de um fluxo de dados.....                   | 36 |
| Figura 7 - Representação de um depósito de dados.....                | 36 |
| Figura 8 - Representação de terminal .....                           | 37 |
| Figura 9 - Diagrama de Fluxo de Dados .....                          | 37 |
| Figura 10 - Modelo de Entidades e Relacionamentos típico .....       | 39 |
| Figura 11- Diagrama de contexto do aplicativo .....                  | 43 |
| Figura 12 - Diagrama de entidade e relacionamento do aplicativo..... | 47 |
| Figura 13 - DFD particionado .....                                   | 48 |
| Figura 14 - Continuação do DFD particionado .....                    | 49 |
| Figura 15 - Matriz CRUD .....  | 50 |
| Figura 16 - Tela de apresentação .....                               | 51 |
| Figura 17 - Tela de entrada do aplicativo .....                      | 52 |
| Figura 18 - Tela de cadastro de multas.....                          | 53 |
| Figura 19 - Tela de cadastro de ruas .....                           | 54 |
| Figura 20 - Tela de cadastro de agentes.....                         | 55 |
| Figura 21 - Tela de cadastro de veículos .....                       | 57 |
| Figura 22 - Tela de cadastro de infrações.....                       | 59 |

|  |    |
|--|----|
| Figura 23 - Tela do aplicativo no SETERB.....  | 60 |
| Figura 24 - Tela de sincronismo de dados ..... | 61 |

# LISTA DE ABREVIATURAS

- CASE** – Computer Aided Software Engineering
- CIASC** – Centro de Informática e Automação do Estado de Santa Catarina
- CNT** – Código Nacional de Trânsito
- CBT** – Código Brasileiro de Trânsito
- DD** – Dicionário de Dados
- DER** – Diagrama de Entidade e Relacionamento
- DFD** – Diagrama de Fluxo de Dados
- EVB** – *Embedded Visual Basic*
- JARI's** – Junta Administrativas de Recurss e Infrações
- PDA's** – Personal Digital Assistant
- SETERB** – Serviço Autônomo Municipal de Terminais Rodoviários de Blumenau
- SAT** – Sistema de Automação de Trânsito
- SNT** – Sistema Nacional De Trânsito



# SUMÁRIO

|  |    |
|--|----|
| 1 INTRODUÇÃO .....                                 | 11 |
| 1.1 OBJETIVO DO TRABALHO .....                     | 13 |
| 1.2 ESTRUTURA DO TRABALHO .....                    | 13 |
| 2 EVOLUÇÃO DOS COMPUTADORES .....                  | 14 |
| 2.1 TIPOS DE COMPUTADORES .....                    | 14 |
| 2.2 MODELOS DE COMPUTADORES .....                  | 16 |
| 2.2.1 SISTEMA OPERACIONAL WINDOWS CE .....         | 19 |
| 2.2.2 APLICAÇÕES .....                             | 21 |
| 3 O AUTOMÓVEL .....                                | 22 |
| 3.1 O TRÂNSITO .....                               | 22 |
| 3.2 CÓDIGO DE TRÂNSITO .....                       | 23 |
| 3.2.1 MUNICIPALIZAÇÃO DO TRÂNSITO .....            | 23 |
| 3.2.1.1 AUTO DE INFRAÇÃO .....                     | 26 |
| 4 TECNOLOGIAS UTILIZADAS .....                     | 28 |
| 4.1 FERRAMENTAS CASE .....                         | 28 |
| 4.2 GENEXUS .....                                  | 30 |
| 4.2.1 DESENHO .....                                | 31 |
| 4.2.2 PROTOTIPAÇÃO .....                           | 32 |
| 4.2.3 PRODUÇÃO .....                               | 33 |
| 4.3 ANÁLISE ESTRUTURADA .....                      | 34 |
| 4.3.1 DIAGRAMA DE FLUXO DE DADOS (DFD) .....       | 35 |
| 4.3.2 DICIONÁRIO DE DADOS (DD) .....               | 38 |
| 4.3.3 FERRAMENTAS PARA ESPECIFICAR PROCESSOS ..... | 38 |

|  |    |
|--|----|
| 4.3.4 MODELO DE ENTIDADES E RELACIONAMENTOS .....  | 38 |
| 4.4 POWER DESIGN .....                             | 40 |
| 5 DESENVOLVIMENTO DO APLICATIVO .....              | 41 |
| 5.1 ESPECIFICAÇÃO DO APLICATIVO .....              | 41 |
| 5.1.1 ANÁLISE DE REQUISITOS .....                  | 42 |
| 5.1.2 DIAGRAMA DE CONTEXTO .....                   | 43 |
| 5.1.3 DICIONÁRIO DE DADOS .....                    | 44 |
| 5.1.3 DIAGRAMA DE ENTIDADE E RELACIONAMENTO .....  | 47 |
| 5.1.4 DIAGRAMA DE FLUXO DE DADOS PARTICIONADO..... | 48 |
| 5.1.5 MATRIZ CRUD.....                             | 50 |
| 5.2 IMPLEMENTAÇÃO DO APLICATIVO .....              | 51 |
| 6 CONCLUSÃO .....                                  | 62 |
| 6.1 DIFICULDADES ENCONTRADAS.....                  | 63 |
| 6.2 SUGESTÕES PARA TRABALHOS FUTUROS .....         | 63 |
| REFERÊNCIAS BIBLIOGRÁFICAS .....                   | 64 |
| ANEXO I.....                                       | 66 |

# 1 INTRODUÇÃO

Quando Henry Ford criou o primeiro automóvel, não tinha em mente a quantidade de veículos que circulariam pelas ruas e tão pouco a diversidade de características das pessoas que conduziriam os veículos. Esta diversidade, tanto de veículos como de motoristas, fez com que fossem criadas normas e regras para condução de automóveis. Leis e normas estas conhecidas como Código de Trânsito. Estas leis e normas são aplicadas através dos agentes de trânsito, que o fazem de forma manual, utilizando um bloco de papel para emitir o auto de infração, quando um motorista as inflige.

Segundo Natale (2002), a automação consiste apenas na aplicação de técnicas que permitem a um conjunto de mecanismos ou equipamentos de processamento de informações executar tarefas antes realizadas pelo homem. Automação não está limitada somente as indústrias, vários setores vem utilizando e descobrindo novas técnicas de automação, dentre os quais os órgãos públicos, no qual trabalham os agentes de trânsito.

De acordo com SETERB (1998) o Serviço Autônomo Municipal de Terminais Rodoviários de Blumenau (SETERB) foi criado pela Lei Municipal nº 2.347, de 27 de março de 1979, como entidade autárquica, subordinada ao Governo Municipal. O SETERB atua como órgão gerenciador e fiscalizador do sistema de trânsito e transporte em Blumenau.

De acordo com SETERB (2003) a municipalização do trânsito é um dos aspectos previstos no novo Código Brasileiro de Trânsito, em vigor desde 1998. Os municípios ficam responsáveis pelos serviços de engenharia, fiscalização, educação de trânsito, Juntas Administrativas de Recursos de Infrações (JARIs) e também pelo levantamento, análise e controle das estatísticas de trânsito. Através de um convênio com a Secretaria de Segurança Pública, os agentes fazem a fiscalização e autuam as infrações cometidas nas vias públicas da cidade. Estes agentes ou guardas de trânsito, como são comumente chamados, possuem um problema: a dificuldade no preenchimento do auto de infração, uma vez que o mesmo é preenchido de forma manual como pode ser visto no anexo I. Muitas vezes com informações prestadas pelo próprio infrator. Desta forma não existe a possível

confiabilidade da confrontação das informações prestadas, sendo que isto pode acarretar na anulação da emissão do auto.

A confrontação das informações prestadas pode ser um auxílio na solução problemática, fazendo-se uso de um equipamento de computação móvel, no qual estarão inseridos todos os dados possíveis sobre os veículos que circulam por determinada região.

O aplicativo será desenvolvido para a automação da emissão dos autos de infrações, visando facilitar o trabalho feito pelos agentes de trânsito, que hoje é feita de forma manual conforme anexo I. O aplicativo pretende auxiliar no trabalho de digitação dos autos, através do sincronismo de dados entre o *hand-held* e o computador no qual servirá de base para envio dos dados ao órgão responsável pela emissão da notificação do infrator. O aplicativo será desenvolvido para ser utilizado em pequenos computadores, chamados *hand-held* ou *palmtops*.

No mercado existem os equipamentos chamados de *hand-helds* ou *palmtops*. São computadores que cabem na palma da mão e podem ser utilizados para a automação do serviço realizado pelos agentes de trânsito, bem como para outras aplicações. A tecnologia *hand-helds*, utilizando pequenos computadores de bolso, pode permitir automação e ganho de produtividade. Este equipamento servirá também de substituto do bloco de papel. Assim tornando mais fácil e ágil a emissão dos autos de infrações de trânsito.

Tem-se como exemplo de utilização de *hand-held*, segundo Palmbr (2003), uma equipe de vendas de uma loja, supermercado ou magazine tem algumas das informações instantâneas, como por exemplo: preços, descontos, posição de estoque entre outros, sem a necessidade de catálogos, listas ou desperdício de tempo procurando preços e informações de produtos pela loja. Para isto, basta digitar o código do produto ou o nome do mesmo e as suas informações aparecem imediatamente na tela.

## 1.1 OBJETIVO DO TRABALHO

O objetivo principal do trabalho proposto visa o desenvolvimento de um aplicativo para o auxílio do agente de trânsito no preenchimento do auto de infração de trânsito.

Os objetivos específicos são:

- a) emitir o auto de infração;
- b) disponibilizar a identificação dos veículos;
- c) disponibilizar lista de infrações;
- d) identificar endereço da autuação;
- e) atualizar dados para envio ao departamento estadual de trânsito.

## 1.2 ESTRUTURA DO TRABALHO

A seguir é apresentada uma síntese dos capítulos constantes desse trabalho.

O capítulo de introdução apresenta uma visão geral do presente trabalho, o contexto em que está inserido, sua importância e objetivo.

O segundo e terceiro capítulos apresentam uma fundamentação do tema, demonstrando uma visão geral do assunto.

O quarto capítulo apresenta as tecnologias e ferramentas utilizadas para o desenvolvimento do aplicativo deste trabalho.

O quinto capítulo apresenta o aplicativo desenvolvido nesse trabalho. Descreve o protótipo de software, suas características, sua especificação, principais telas.

O sexto capítulo apresenta as conclusões desse trabalho e as sugestões para que o mesmo possa ter continuidade e seja melhorado.

## 2 EVOLUÇÃO DOS COMPUTADORES

Não há dúvida alguma de que as pessoas estão se tornando cada vez mais móveis, não só em onde elas vivem e trabalham, mas em como elas vivem e trabalham. A portabilidade tem se tornado um dos pontos de venda mais fortes no mercado atual, de cosméticos, vídeo games, aparelhos de CD de bolso a *paggers*, telefones móveis e computadores portáteis.

O mundo real está se tornando parecido com a história em quadrinhos em uma velocidade assustadora e dando mais força computacional realmente personalizada do que os autores dos quadrinhos poderiam imaginar ser possível. O poder de computação massivo está sendo armazenado em dispositivos cada vez menores.

De acordo com Bizzotto (1998), o primeiro computador eletrônico: o ENIAC ocupava uma área superior a 170 metros quadrados, pesava 30 toneladas, utilizando 18.000 válvulas e 10.000 capacitores, consumindo 150.000 watts de energia e custava alguns milhões de dólares. Atualmente, os microcomputadores pesam alguns quilos e possuem capacidade superior ao ENIAC. Hoje existem a disposição computadores que cabem na palma da mão.

### 2.1 TIPOS DE COMPUTADORES

Abaixo descreve-se os tipos de computadores existentes:

- Computador pessoal: Um computador pequeno, relativamente barato e projetado para um usuário individual. Todos são baseados na tecnologia do microprocessador que habilita os fabricantes a colocarem toda uma UCP em um único chip. Os computadores pessoais são utilizados para processamento de textos, editoração, execução de planilhas de cálculo e aplicações de gerenciamento de base de dados. Em casa, o uso mais popular para computadores pessoais é para jogar e acessar a Internet. Os computadores pessoais apareceram no final dos anos 70. Um dos primeiros e mais popular computadores pessoais foi o Apple II, introduzido em 1.977 pela *Apple Computer*;

- *Workstation*: (estação de trabalho): um poderoso computador pessoal. Uma *workstation* é parecida com um computador pessoal, mas tem um microprocessador mais poderoso e um monitor de mais alta qualidade e trabalha ligado a um computador mais poderoso, a uma rede, ou ambos. É um tipo de computador utilizado para aplicações de engenharia editoração, desenvolvimento de software e outros tipos de aplicações que requeiram uma quantidade moderada de poder de computação e capacidades gráficas relativamente altas;
- Minicomputador: um computador médio. Em tamanho e poder os minicomputadores se encontram entre *workstations* e *mainframes*. Na década passada, a distinção entre grandes minicomputadores e pequenos *mainframes* ficou confusa tal qual a distinção entre pequenos minicomputadores e *workstations*. Mas, em geral, um minicomputador é um sistema multiprocessado capaz de suportar entre 4 a 200 usuários simultaneamente com um grau aceitável de performance e satisfação dos usuários;
- *Mainframe*: um computador muito grande e caro capaz de suportar centenas ou mesmo milhares de usuários simultaneamente e também de executar muitos programas ao mesmo tempo. Na hierarquia que se inicia com um simples microprocessador (em relógios, por exemplo) e vai até supercomputadores no topo, os *mainframes* estão logo abaixo dos supercomputadores. Em alguns casos, os *mainframes* são até mais poderosos que os supercomputadores porque suportam mais programas simultâneos. Mas os supercomputadores podem executar um único programa mais rápido que um *mainframe*;
- Supercomputador: o mais rápido dos computadores. Os supercomputadores são muito caros e são empregados para aplicações especializadas que demandem quantidades imensas de cálculos matemáticos. Por exemplo, a previsão do tempo exige o uso de um supercomputador. Outros usos do

supercomputador incluem gráficos animados, cálculo da dinâmica dos fluídos, pesquisa em energia nuclear e exploração de petróleo. A diferença principal entre um supercomputador e um *mainframe* é que o primeiro canaliza todo o seu poder a fim de executar alguns programas tão rápido quanto possível, enquanto um *mainframe* utiliza seu poder para executar muitos programas concorrentemente e atender muitos usuários.

## 2.2 MODELOS DE COMPUTADORES

Os computadores pessoais ou *workstations* podem apresentar várias formas, cada uma de acordo com o tipo de função que venham a desempenhar. É necessário escolher o modelo correto para cada tipo de trabalho que se venha a desempenhar. Os modelos que mais se destacam na atualidade são os seguintes:

- Modelo *desktop*: um computador projetado para caber confortavelmente sobre uma mesa, tipicamente com o monitor permanecendo sobre o computador. Os modelos *desktop* são largos e baixos, enquanto os modelo torre são estreitos e altos. Por causa da sua forma, os computadores modelo *desktop* são, geralmente, limitados a três dispositivos de armazenamento de massa. Existem modelos *desktop* projetados para ser muito pequenos e são chamados de *modelos slimline*;
- Modelo torre: refere-se a um computador em que a fonte de alimentação, a placa mãe e os dispositivos de armazenamento de massa são empilhados um sobre o outro no gabinete. Este é, o contrário do modelo *desktop*, em que estes componentes são dispostos em uma caixa mais compacta. A grande vantagem dos modelos torre é que existem poucas restrições de espaço, o que torna a instalação de dispositivos adicionais de armazenamento muito mais fácil;
- Modelo *notebook*: um tipo de computador pessoal extremamente leve. Os *notebooks* pesam, via de regra, menos que três quilos e são pequenos o bastante para caber facilmente dentro de uma bolsa portátil. Afora o



tamanho, a principal diferença entre um *notebook* e um computador pessoal comum (*desktop* ou torre) é o monitor. Os *notebooks* utilizam uma variedade de técnicas conhecidas como tecnologias de *flat pannel* (painel chato) que geram monitores leves e não volumosos. Em termos de poder de computação, os *notebooks* modernos estão quase emparelhados com os computadores pessoais comuns. Eles têm as mesmas CPUs, capacidade de memória e discos. Contudo, todo este poder em uma pequena caixa é caro;

- Computador *hand-held*: um computador portátil que é pequeno o suficiente para caber em uma mão. Embora extremamente conveniente de carregar, os computadores *hand-held* não substituíram os *notebooks* por causa de seus teclados e telas pequenos. Os computadores *hand-held* mais populares são aqueles projetados especificamente para fornecer funções de gerenciamento de informações pessoais (PIM – *Personal Information Management*), tais como calendário, lista de endereços, agendas, processador de textos, planilhas, etc;

Figura 1 - Exemplo de um computador *hand-held*



Fonte : Palmland (2003?)

- *Palmtop*: um pequeno computador que literalmente cabe na palma da sua mão. Comparados a computadores de tamanho normal, os *palmtops* são muito limitados, mas são práticos por terem certas funções tais como

agendas telefônicas e calendários. *Palmtops* e *hand-helds* podem ser sinônimos, mas uma diferença está se acentuando. Enquanto os *hand-helds* têm teclado, os *palmtops* não possuem um teclado físico. Este aparece na tela de cristal líquido e cada tecla é pressionada com a caneta ótica. Os *palmtops* possuem *drives* de disco, *modems* e outros dispositivos mais sofisticados. Os *palmtops* e *hand-helds* são denominados computadores de bolso.

Figura 2 - Exemplo de um computador *Palmtop*



Fonte : Palmland(2003?)

O que é interessante sobre o campo dos *hand-helds* é que estes pequenos computadores de uso geral estão se fundindo com uma classe de *hand-helds* chamados de assistentes digitais pessoais (*Personal Digital Assistants* ou PDA's). O propósito dos PDA's é serem assistentes inteligentes, capazes de armazenar e recuperar nomes e endereços, notas de fax e até mesmo reconhecer letras escritas a mão e rabiscos de caneta. A singularidade destas máquinas é representada pela sua inteligência. Em vez de simplesmente executar as tarefas requisitadas pelo usuário, estes dispositivos serão capazes de realizar séries de comandos e antecipar ações antes que elas sejam pedidas. Por exemplo, ele pode formatar um rascunho em uma carta comercial, fornecendo o nome completo e endereço, saudação e formatação.

Faz algum tempo que a informática cabe num bolso. Precisamente, os responsáveis disto são sistemas como os PDA's e os *hand-held* ou HPC. Basicamente, trata-se de autênticos computadores que lhe permitem levar consigo, esteja onde estiver, seus dados

mais importantes. Os sistemas PDA's e *hand-held* são a vanguarda do que denomina-se como informática de bolso. Trata-se de verdadeiros computadores pessoais que cabem na palma da mão, com enormes prestações e a um preço cada vez mais baixo.

Outras qualidades que os fazem diferentes são o tamanho e o peso. Obviamente, a renúncia ao teclado determina que os PDA's sejam menores, de fato parecem simplesmente uma tela, e mais leves, pelo que resultam mais cômodos já que, em muitos casos, não são muito maiores que um maço de cigarros. Por sua parte, os *hand-helds* são algo maiores, não muito, porém, o suficiente para que não entrem no bolso da sua camisa.

*Pocket PC* e *hand-held PC* referem-se especificamente aos tipos de aparelhos portáteis ou PC's de bolso, enquanto *Windows CE*, é o sistema operacional que trabalha no coração destes aparelhos.

Abaixo descreve-se algumas características destes aparelhos:

- o *Pocket PC* é um tipo de aparelho em que a tela mede 1/4 de VGA (320x240) e contêm um pacote consistente de aplicativos integrados, opções de acesso à redes com ou sem fio;
- o *hand-held PC* é um tipo de aparelho em que a tela pode medir metade de VGA (640x240) ou inteiro (640x480 ou 800x600) com ou sem teclado embutido e contêm um consistente pacote de aplicativos integrados, opções de acesso à redes com ou sem fio.

## 2.2.1 SISTEMA OPERACIONAL WINDOWS CE

De acordo com Michaelis (2003?), *hand held* é um computador muito pequeno que pode ser segurado na mão, útil para a entrada de informação básica, quando não existe um terminal disponível.

*Hand holds* são computadores de mão que não fazem uso de um sistema operacional específico *WindowsCE*. Têm tamanho reduzindo podendo chegar a um tamanho de uma fita cassete de vídeo e diferenciam-se dos *palmtop's* por possuírem teclado.

De acordo com Palmland (2003?), os *hand held* são divididos em duas categorias:

- *Handheld PC*;
- *Handheld PC Pro*.

Os *Handhelds PC* possuem o tamanho de uma fita de vídeo e sua tela geralmente não excede os 17 cm de diagonal. São recomendados para anotações rápidas e leitura e envio de e-mail. Pode possuir saída para VGA, *slots compact Flash*, fax modem interno entre outros dispositivos e utilizam o sistema operacional *Windows CE H/PC*.

Já os *Handhelds PC Pros* são produtos maiores e geralmente possuem telas de mais de 20 cm diagonal. São ótimos aliados quando é necessária digitação constante e/ou trabalho mais árduo. Estes equipamentos suportam conexões USB, *Hard Disks* tipo II e utilizam o sistema operacional *Windows CE H/PC PRO*.

O sistema operacional roda diretamente na memória ROM e possui características *real time*, que garante velocidade. O sistema não necessita carregar *drivers* e quando o *palmtop* é ativado, o *Windows CE* é inicializado automaticamente. Pode-se desligar o *palmtop* a qualquer momento, independente da aplicação que está em uso. Não há função de desligar o *palmtop* via software como ocorre com o sistema operacional *Windows* dos computadores pessoais. O *Windows CE* é um sistema operacional embutido e, portanto não há como fazer *upgrade* sem a troca do chip-mãe.

Ele é fornecido gratuitamente nos equipamentos e é dividido em três categorias:

- ***Windows CE P/PC*** - Utilizado nos *Palmtops*;
- ***Windows CE H/PC*** - Utilizados nos *hand-helds*;
- ***Windows CE H/PC Pro*** - Utilizado nos *hand-helds* profissionais.

De acordo com Palmland (2003?), estes equipamentos utilizam geralmente o sistema operacional *Windows CE* ou *Palm OS*, que permitem uma imensa gama de opções tanto em software como em multimídia. São muito práticos na vida profissional e podem ser um excelente companheiro para diversão, pois os últimos modelos já incorporam fone de ouvido onde se podem escutar músicas com qualidade de CD utilizando o formato MP3.

Agendas de números e endereços, anotações rápidas, gravador de sons, alto falantes, tela colorida de alta resolução são as maiores diferenças para uma agenda eletrônica normal.

## 2.2.2 APLICAÇÕES

Os *hand-helds* e *PocketPCs* são equipamentos portáteis que podem ser utilizados para diversas aplicações de computação, melhorando a produtividade de empresas tais como:

- companhias aéreas: onde podem ser oferecidas informações sobre horários de vôos, programas de milhagens, etc;
- força de vendas: os vendedores podem levar as informações sobre seus clientes sempre atualizadas, fazer orçamentos e fechar pedido;
- institutos de pesquisas: os formulários de pesquisado podem ser substituídos por programas específicos para cada pesquisa, não tendo mais a necessidade de digitalização das informações pesquisadas;
- controle de estoques/logística: para esta situação temos o acompanhamento de produtos, tais como: localização, estoque, movimentação dos mesmos dentro de uma empresa;
- órgão públicos: onde podem ser utilizados para automação de serviços manuais, como: emissão de autos de infrações de trânsito, leitores de consumo de energia elétrica ou água consumida.

## 3 O AUTOMÓVEL

De acordo com Billian (1975), com a invenção da máquina a vapor por James Watt em 1778 foi criada a possibilidade de substituir a força do cavalo para o transporte sobre rodas pela força mecânica. Entretanto, só em 1875, surgiu pela primeira vez um pequeno carro de madeira com motor a combustão feito pelo construtor Siegfried Markus e em 1885, Carl Benz construiu um carro a motor. A fabricação em série dos primeiros automóveis teve início em 1896, ano em que Henry Ford lançou seu primeiro carro, o “*Quadricycle*”. No ano de 1903, Henry Ford pôs em prática sua linha de montagem.

A primeira viagem de automóvel foi realizada em 1908 pelo Conde Lesdain. Ele levou trinta e seis dias para ir do Rio de Janeiro a São Paulo. Atualmente, um avião faz o mesmo percurso em trinta minutos e um automóvel, numa velocidade média de 80 km/hora, demora aproximadamente 5 horas.

A indústria automobilística teve uma evolução considerável dos anos iniciais até o presente. Evolução que pode ser vista através dos modelos fabricados hoje em dia, onde tem-se a aplicação cada vez maior de tecnologias, tanto para a fabricação em si como um diferencial entre os modelos fabricados.

### 3.1 O TRÂNSITO

De acordo com o Código de Trânsito Brasileiro (CTB), publicado em 24 de setembro de 1998, no artigo 1º, parágrafo 1º, considera-se trânsito a utilização das vias por pessoas, veículos e animais, isolados ou em grupos, conduzidos ou não, para fins de circulação, parada, estacionamento e operação de carga e descarga.

No artigo 5º. do referido código, temos a definição do Sistema Nacional de Trânsito (SNT), no qual nos relata as composições e competências de cada órgão responsável pelo trânsito em seu âmbito geográfico.

De acordo com o CTB (1998), no artigo 8º., os estados, o Distrito Federal e os municípios organizarão os respectivos órgãos e entidades executivos de trânsito e executivos rodoviários, estabelecendo os limites circunscricionais de suas atuações.

## **3.2 CÓDIGO DE TRÂNSITO**

No ano de 1941 foi publicado o primeiro Código Nacional de Trânsito (CNT). De acordo com SETERB(1998) o novo Código de Trânsito Brasileiro é a Lei 9.503, de 23/09/97 e foi publicada no Diário Oficial da União em 24 de Setembro de 1997. A Lei possui 341 artigos que proporcionam instrumentos e condições para que o processo de circulação de bens e pessoas através do espaço físico brasileiro, tanto rural como urbano, se desenvolva dentro de padrões de segurança, eficiência, fluidez e conforto. O Código foi sancionado no dia 23 de Setembro de 1997. O artigo 340 define que a lei entraria em vigor 120 dias após a sua publicação. Como ele foi publicado no Diário Oficial em 24 de setembro, o CTB entrou em vigor no dia 22 de janeiro de 1998.

Segundo DENATRAN (2003) o Código de Trânsito Brasileiro, prevê uma clara divisão de responsabilidades e uma sólida parceria entre órgãos federais, estaduais e municipais. Os municípios, em particular, tiveram sua esfera de competência substancialmente ampliada no tratamento das questões de trânsito. Por isso, compete agora aos órgãos executivos municipais de trânsito exercer nada menos que vinte e uma atribuições. Uma vez preenchidos os requisitos para integração do município ao SNT(Sistema Nacional de Trânsito), ele assume a responsabilidade pelo planejamento, o projeto, a operação e a fiscalização, não apenas no perímetro urbano, mas também nas estradas municipais. A prefeitura passa a desempenhar tarefas de sinalização, fiscalização, aplicação de penalidades e educação de trânsito.

### **3.2.1 MUNICIPALIZAÇÃO DO TRÂNSITO**

De acordo com SETERB (2003) a municipalização do trânsito é um dos aspectos previstos no novo Código Brasileiro de Trânsito, em vigor desde 1998. Por ele, os municípios ficam responsáveis pelos serviços de engenharia, fiscalização, educação de

trânsito, Juntas Administrativas de Recursos de Infrações (JARIs) e também pelo levantamento, análise e controle das estatísticas de trânsito das cidades.

De acordo com SETERB (2003) o Serviço Autônomo Municipal de Terminais Rodoviários de Blumenau (SETERB) foi criado pela Lei Municipal nº 2.347, de 27 de março de 1979, como entidade autárquica, subordinada ao Governo Municipal, dispendo de autonomia econômico-financeira dentro dos limites traçados na lei.

O SETERB atua como órgão gerenciador e fiscalizador do sistema de trânsito e transporte em Blumenau. No início tinha suas ações concentradas no Terminal Rodoviário Hercílio Deeke, passando depois a criar e aprimorar o sistema de atendimento aos usuários e a administrar o crescente fluxo de veículos nas ruas, principalmente na região central da cidade.

De acordo com SETERB (1998), o artigo 24 do CTB descreve as obrigações dos órgãos municipais responsáveis pelo trânsito. A seguir transcrevemos o artigo para melhor compreensão.

Art. 24. Compete aos órgãos e entidades executivos de trânsito dos municípios, no âmbito de sua circunscrição:

I - cumprir e fazer cumprir a legislação e as normas de trânsito, no âmbito de suas atribuições;

II - planejar, projetar, regulamentar e operar o trânsito de veículos, de pedestres e de animais, e promover o desenvolvimento da circulação e da segurança de ciclistas;

III - implantar, manter e operar o sistema de sinalização, os dispositivos e os equipamentos de controle viário;

IV - coletar dados estatísticos e elaborar estudos sobre os acidentes de trânsito e suas causas;

V - estabelecer, em conjunto com os órgãos de polícia ostensiva de trânsito, as diretrizes para o policiamento ostensivo de trânsito;



VI - executar a fiscalização de trânsito, autuar e aplicar as medidas administrativas cabíveis, por infrações de circulação, estacionamento e parada previstas neste Código, no exercício regular do Poder de Polícia de Trânsito;

VII - aplicar as penalidades de advertência por escrito e multa, por infrações de circulação, estacionamento e parada previstas neste Código, notificando os infratores e arrecadando as multas que aplicar;

VIII - fiscalizar, autuar e aplicar as penalidades e medidas administrativas cabíveis relativas a infrações por excesso de peso, dimensões e lotação dos veículos, bem como notificar e arrecadar as multas que aplicar;

IX - fiscalizar o cumprimento da norma contida no art. 95, aplicando as penalidades e arrecadando as multas nele previstas;

X - implantar, manter e operar sistema de estacionamento rotativo pago nas vias;

XI - arrecadar valores provenientes de estada e remoção de veículos e objetos, e escolta de veículos de cargas superdimensionadas ou perigosas;

XII - credenciar os serviços de escolta, fiscalizar e adotar medidas de segurança relativas aos serviços de remoção de veículos, escolta e transporte de carga indivisível;

XIII - integrar-se a outros órgãos e entidades do Sistema Nacional de Trânsito para fins de arrecadação e compensação de multas impostas na área de sua competência, com vistas à unificação do licenciamento, à simplificação e à celeridade das transferências de veículos e de prontuários dos condutores de uma para outra unidade da Federação;

XIV - implantar as medidas da Política Nacional de Trânsito e do Programa Nacional de Trânsito;

XV - promover e participar de projetos e programas de educação e segurança de trânsito de acordo com as diretrizes estabelecidas pelo CONTRAN;

XVI - planejar e implantar medidas para redução da circulação de veículos e reorientação do tráfego, com o objetivo de diminuir a emissão global de poluentes;

XVII - registrar e licenciar, na forma da legislação, ciclomotores, veículos de tração e propulsão humana e de tração animal, fiscalizando, autuando, aplicando penalidades e arrecadando multas decorrentes de infrações;

XVIII - conceder autorização para conduzir veículos de propulsão humana e de tração animal;

XIX - articular-se com os demais órgãos do Sistema Nacional de Trânsito no Estado, sob coordenação do respectivo CETRAN;

XX - fiscalizar o nível de emissão de poluentes e ruído produzidos pelos veículos automotores ou pela sua carga, de acordo com o estabelecido no art. 66, além de dar apoio às ações específicas de órgão ambiental local, quando solicitado;

XXI - vistoriar veículos que necessitem de autorização especial para transitar e estabelecer os requisitos técnicos a serem observados para a circulação desses veículos.

§ 1º As competências relativas a órgão ou entidade municipal serão exercidas no Distrito Federal por seu órgão ou entidade executivos de trânsito.

§ 2º Para exercer as competências estabelecidas neste artigo, os Municípios deverão integrar-se ao Sistema Nacional de Trânsito, conforme previsto no art. 333 deste código.

De acordo com SETERB (1998), criada em 1955, a Guarda de Trânsito de Blumenau é a mais antiga do Brasil. Através de um convênio com a Secretaria de Segurança Pública, os agentes fazem a fiscalização e autuam as infrações cometidas nas vias públicas da cidade.

### **3.2.1.1 AUTO DE INFRAÇÃO**

Descreve-se abaixo uma visão macro do processo de preenchimento de um auto de infração e seu tramite até o envio do mesmo ao CIASC (Centro de Informática e Automação do Estado de Santa Catarina), órgão responsável pela manipulação dos dados referentes ao auto de infração.

Ao emitir o auto de infração, o agente de trânsito o faz em três vias, sendo elas destinadas da seguinte maneira:

- a primeira via destina-se ao setor de digitação do auto;
- a segunda via destina-se ao infrator e
- a terceira via destina-se ao arquivo.

Ao final de seu expediente o agente de trânsito faz o recolhimento dos autos de infrações por ele emitidos e destaca do bloco as vias destinadas a digitação. As vias são depositadas em uma urna, para em seguida serem digitadas no sistema que fará o envio ao CIASC.

O aplicativo tem como um de seus objetivos eliminar esta digitação, através do sincronismo entre o *palmtop* e o PC e conseqüentemente fazer o envio dos autos automaticamente.

## 4 TECNOLOGIAS UTILIZADAS

Para a realização deste trabalho foram utilizadas algumas ferramentas e tecnologias, a fim de especificar a análise e desenvolver o aplicativo. Neste capítulo são abordadas as análise estruturada, a ferramenta *CASE Power Design* e a ferramenta de programação Genexus.

### 4.1 FERRAMENTAS CASE

A necessidade de se produzir algo com mais velocidade e eficiência, é vista em todos os setores da produção, inclusive na área da computação. As ferramentas *Computer Aided Software Engineering* (CASE) surgiram da necessidade de se ganhar velocidade e agilidade no processo de desenvolvimento e de se ter as informações centralizadas e organizadas de maneira lógica.

Uma ferramenta CASE é um produto computacional que suporta uma ou mais das atividades do processo de software. A introdução dessas ferramentas visa melhorar a qualidade do software e aumentar a produtividade do seu processo de produção. As ferramentas CASE podem ser :

- horizontais - oferecem serviços utilizados durante todo o processo de software, tais como suporte à documentação e gerenciamento de versões e configurações;
- verticais - são utilizadas em fases específicas do processo de software, tais como análise de requisitos e teste de software.

De acordo com Pressmann (1995) as ferramentas CASE podem ser classificadas por função, por seus papéis como instrumentos para gerentes e para o pessoal técnico, pelo uso que elas têm nas várias etapas do processo de engenharia de software, pela arquitetura do ambiente (*hardware* e software) que as suporta ou até mesmo pela origem ou custo delas.

Uma proposta de classificação apresentada por Pressman (1995) é descrita abaixo. Através desta classificação observa-se o amplo espectro de ferramentas CASE existentes,

apesar de ser comum a referência a ferramentas CASE como ferramentas específicas para análise e projeto de software.

- planejamento de sistemas gerenciais;
- gerenciamento de projetos;
- especificação de requisitos;
- especificação formal de sistemas;
- documentação;
- comunicação;
- controle de qualidade;
- gerenciamento de versões e configurações;
- análise e projeto de software;
- projeto e desenvolvimento de interfaces;
- programação.

Utilizou-se de ferramentas CASE para programação, que subdivide-se em:

- ferramentas de codificação convencionais;
- ferramentas de codificação de quarta geração;
- ferramentas de programação orientadas a objeto.

Especificamente, neste trabalho foi utilizada a ferramenta de codificação Genexus.

## 4.2 GENEXUS

A ferramenta de desenvolvimento Genexus é considerada uma ferramenta incremental. De acordo com Artech (1992), Genexus é uma ferramenta baseada no conhecimento, cujo objetivo é auxiliar o analista de sistema a implementar aplicações no menor tempo e com maior rapidez possível e auxiliar os usuários durante todo o ciclo de vida das aplicações.

De acordo com Junior (2000), abaixo são mostradas algumas características da ferramenta Genexus:

- desenho automático e criação da base de dados;
- geração e manutenção automática dos programas de aplicação;
- prototipação integral das aplicações em microcomputador;
- desenvolvimento de um único protótipo, independente da plataforma de produção;
- linguagem complementar procedural;
- distribuição do conhecimento corporativo, visando facilitar o desenvolvimento de novas aplicações;
- verificação da consistência e consolidação entre as aplicações desenvolvidas separadamente.

O processo de desenvolvimento e manutenção da aplicação é bastante simplificado quando realizado pela ferramenta Genexus. O ciclo de vida de uma aplicação desenvolvida em Genexus é composto de três fases:

- desenho;
- prototipação;
- produção.

De acordo com Artech (1992), a ferramenta Genexus proporciona muitos benefícios para os desenvolvedores de aplicações, aos quais cita-se abaixo:

- os usuários, participantes do processo de desenvolvimento, ao sugerirem modificações no protótipo podem ver o impacto que as mesmas causarão rapidamente;
- a automação no desenvolvimento, manutenção e documentação aumenta a produtividade;
- as aplicações desenvolvida em Genexus podem operar tanto em plataforma PC, como em AS/400 ou em máquinas Risc.

### **4.2.1 DESENHO**

De acordo com Dias (2002), o desenho da base de dados é um processo determinístico: dado um conjunto de objetos do usuário, existe uma única base de dados relacional mínima que o satisfaz. As bases de dados que a ferramenta Genexus desenha estão na terceira forma normal e têm índices que são estritamente necessários.

Nesta fase é realizada a análise do sistema. É também nesta fase que são descritas as visões dos usuários, as estruturas dos dados, os atributos, as fórmulas e as regras de trabalho. A ferramenta Genexus utiliza-se de cinco objetos para desenhar uma aplicação:

- transações: modificações interativas, que permitem ao usuário criar, modificar ou eliminar informações da base de dados;
- *work panels*: traduzidos por painéis de trabalho, são telas onde o usuário pode fazer consultas interativas a base de dados;
- relatórios: conforme Artech (1992), um relatório é um processo que permite visualizar as informações contidas na base de dados. A listagem é feita de forma convencional, através da visualização em tela ou em papel;
- procedimentos: processos em que o usuário pode inserir, alterar e excluir dados na base de dados;

- *web objects*: conforme Dias (2002), possuem as mesmas características das *work panels*. Permitem criar páginas *WEB* dinâmicas com as quais se implementam os diálogos necessários e chamando procedimentos, permitem alterações na base de dados.

## 4.2.2 PROTOTIPAÇÃO

O protótipo é importante por que permite que toda aplicação seja testada antes de se passar à produção. Durante os testes o usuário pode trabalhar com dados reais, prever alterações necessárias a adequação correta do sistema.

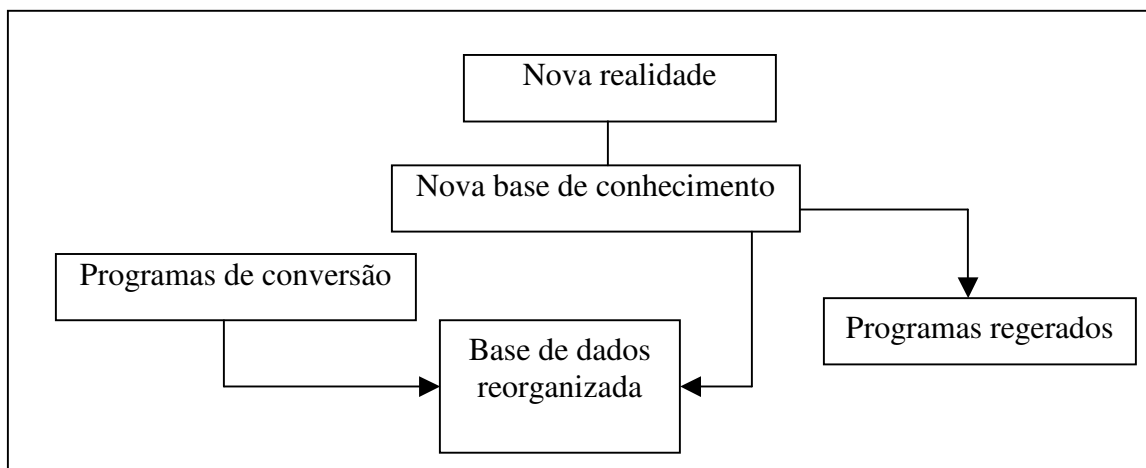
Descreve-se abaixo algumas vantagens da prototipação:

- permite ver resultados de forma imediata;
- permite a entrega parcelada dos requerimentos do usuário;
- detecção de erros de forma rápida;
- cumpre o compromisso dos usuários com o desenvolvimento;
- sistemas de melhor qualidade.

A ferramenta Genexus utiliza a filosofia do desenvolvimento incremental. Na ferramenta Genexus as trocas no projeto durante a implementação não são desgastantes, pois ele constrói uma aplicação com uma metodologia de aproximações sucessivas, sem custo adicional. Este procedimento é mostrado na Figura 3:



Figura 3 - Impacto da alteração da realidade



Fonte: Adaptado de Junior (2000)

### 4.2.3 PRODUÇÃO

De acordo com Junior (2000), a diferença existente na prototipação e produção é que a prototipação é desenvolvida em um ambiente de microcomputador, enquanto a produção é realizada no ambiente desejado pelo usuário.

A ferramenta Genexus trabalha com os seguintes ambientes:

- IBM OS/2, AT, XT, PC;
- IBM AS/400;
- rede de microcomputadores;
- processamento corporativo entre microcomputadores e AS/400.

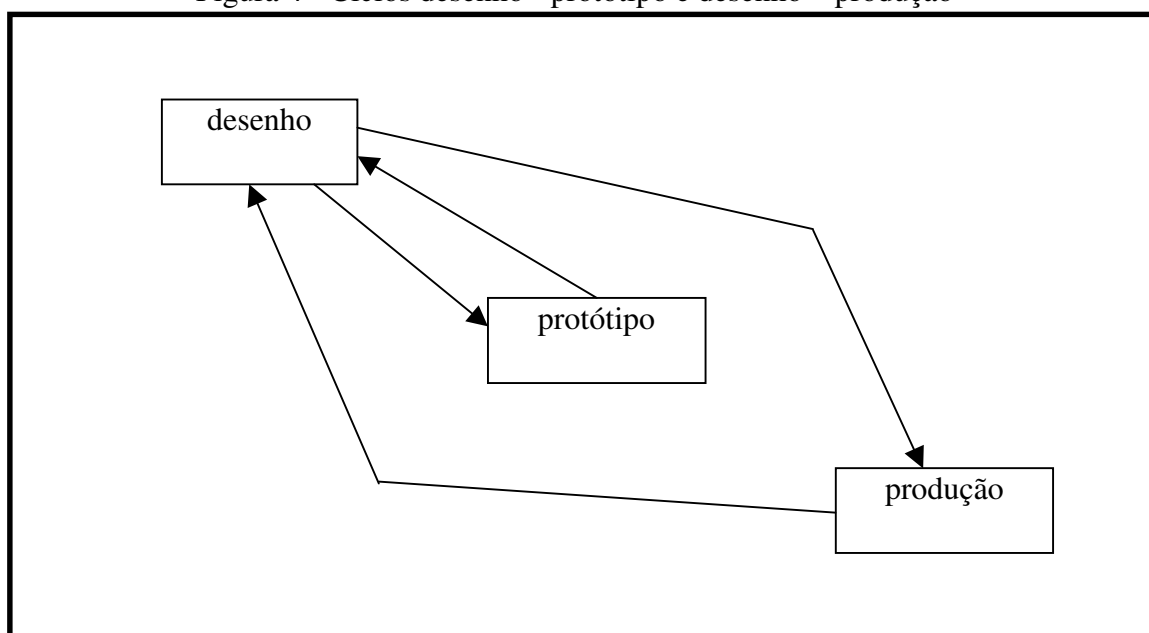
O analista trabalha com inteira liberdade em um ambiente de prototipação, com uma pequena base de conhecimento e, quando a aplicação estiver pronta é incorporada em outras aplicações (JUNIOR, 2000).

Utilizando a ferramenta Genexus, a tarefa básica do analista é a descrição da realidade. A partir do momento que isto modifica a atividade do analista e, também, o de seu perfil de otimização, isto o transforma em um analista de negócios. Agora trabalha em alto nível, discutindo problemas com o usuário e testando com ele as especificações a nível

de protótipo, em vez de desenvolver sua atividade e através de tarefas de baixo nível como, desenhar arquivos, normalizar, desenhar programas, programar, buscar e eliminar os erros dos programas.

A partir da base de conhecimento, a ferramenta Genexus gera automaticamente tanto os programas de criação / reorganização da base de dados como os da aplicação. Na Figura 4, observam-se as fases do ciclo de desenvolvimento, quando utiliza-se da ferramenta Genexus.

Figura 4 - Ciclos desenho - protótipo e desenho – produção



Fonte: Adaptado de Artech (1992).

## 4.3 ANÁLISE ESTRUTURADA

Segundo Gane (1984), análise estruturada é a construção de um modelo lógico de um sistema, utilizando técnicas gráficas capazes de levar usuários, analistas e projetistas a formarem um quadro claro e geral do sistema e de como suas partes se encaixam para atender às necessidades daqueles que dele precisam.

É composta de um conjunto de técnicas e ferramentas em constante evolução, nascido do sucesso da programação e do projeto estruturados (GANE, 1984).

Segundo Demarco (1989), projeto estruturado é uma estratégia para a produção de projeto *top-down*, com alta capacidade de manutenção facilmente testável. O projeto estruturado começa com as características hierárquicas, fazendo com que você trate as questões em ordem de importância, ao invés de tratá-las na ordem em que o computador irá encará-las.

De acordo com Yourdon (1990), usando a análise estruturada, o usuário adquire um entendimento claro do sistema que está sendo especificado e o projetista pode criar um projeto estruturado mais rapidamente e mais acurado. A análise estruturada possui alguns componentes básicos que definem sua estrutura:

- diagrama de fluxo de dados (DFD);
- dicionário de dados (DD);
- ferramentas para especificar processos;

### **4.3.1 DIAGRAMA DE FLUXO DE DADOS (DFD)**

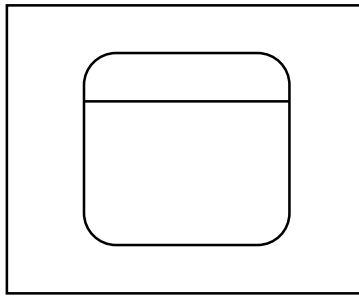
Segundo Yourdon (1993), o diagrama de fluxo de dados (DFD), identifica as funções do sistema e como elas utilizam as informações e as transferem entre as funções existentes.

Segundo Demarco (1989), o DFD é composto por quatro elementos básicos:

- a) fluxo de dados, representados por flechas;
- b) processos, representados por círculos;
- c) arquivos, representados por linhas retas;
- d) fontes ou destinos de dados, representados por caixa.

O processo é um componente procedural do sistema, operando sobre os dados, podendo executar operações aritméticas ou lógicas com os dados para produzir algum resultado, devendo ser usado um nome significativo para definir a operação executada pelo processo (Martin 1991). Processos são representados como círculos ou “bolhas” no diagrama, representam as diversas funções individuais que o sistema executa. Funções transformam entradas em saídas.

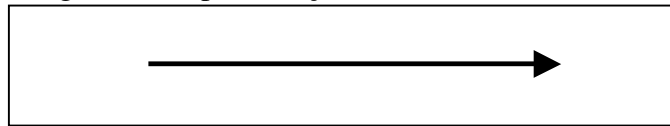
Figura 5 - Representação de um processo



Fonte: Adaptado de Yourdon (1990)

Fluxos são mostrados pelas setas direcionais. Elas são as conexões entre os processos e representam a informação que os processos exigem como entrada e/ou as informações que eles geram como saída:

Figura 6 - Representação de um fluxo de dados

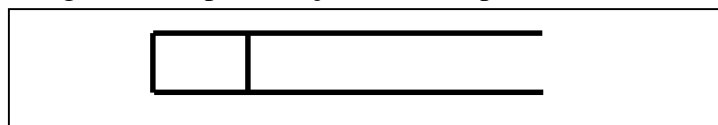


Fonte: Adaptado de Yourdon (1990)

O arquivo, ou depósito de dados são repositórios de dados produzido por um processo, preservados por um determinado período, e então utilizados por um outro processo (JOÃO, 1993).

Depósitos de dados são representados por duas linhas paralelas ou por uma elipse. Eles mostram coleções de dados que o sistema deve manter por determinado período:

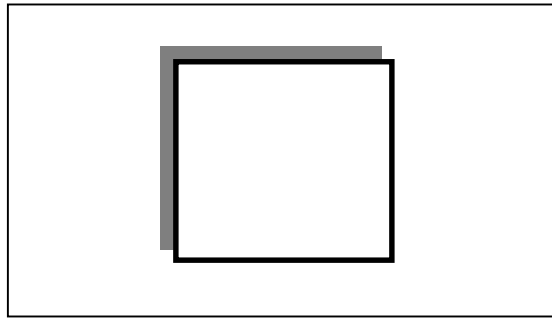
Figura 7 - Representação de um depósito de dados



Fonte : Adaptado de Yourdon (1990)

Terminais mostram as entidades externas com as quais o sistema se comunica. Os terminais são tipicamente, indivíduos, grupos de pessoas (por exemplo, um outro departamento ou divisão da organização), outros sistemas e organizações externas. Na Figura 8, tem-se um exemplo de um terminal.

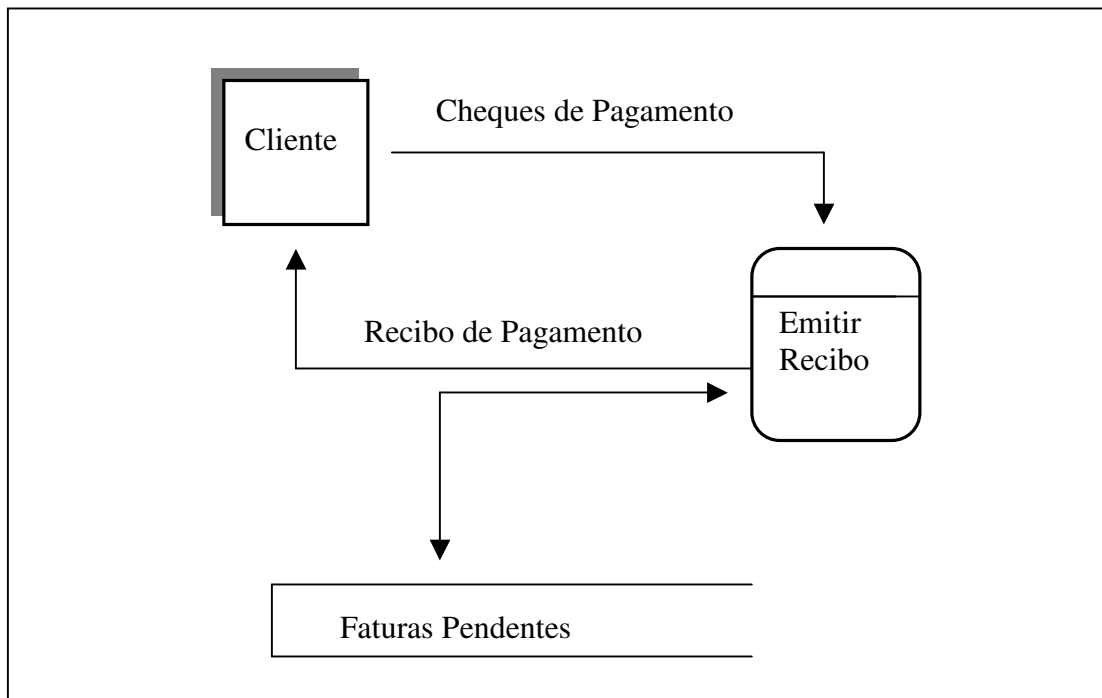
Figura 8 - Representação de terminal



Fonte: Adaptado de Yourdon (1990)

Na Figura 9 temos um exemplo de um DFD. Neste exemplo existe uma entidade externa chamada Cliente que envia cheque de pagamento e um depósito de dados chamado Faturas Pendentes, que é consultado, quando da chegada de um cheque para pagamentos, para verificar se a fatura está registrada. Conforme a descrição do evento, há uma resposta externa (no caso, um fluxo de dados em direção á entidade externa Cliente):

Figura 9 - Diagrama de Fluxo de Dados



Fonte : Adaptado de Yourdon (1990)

### **4.3.2 DICIONÁRIO DE DADOS (DD)**

De acordo com Yourdon (1990), embora o DFD (diagrama de fluxo de dados) ofereça uma visão geral dos principais componentes funcionais do sistema, não fornece qualquer detalhe sobre estes componentes. Para mostrar detalhes de qual informação é transformada e como é transformada, são necessárias duas ferramentas de suporte textual de modelagem: o dicionário de dados e a especificação do processo.

Conforme Pompilho (1994) um dicionário de dados é um repositório de informações sobre os componentes dos sistemas.

### **4.3.3 FERRAMENTAS PARA ESPECIFICAR PROCESSOS**

De acordo com Yourdon (1990), existem diversas ferramentas que podem ser utilizadas para produzir uma especificação de processos: tabelas de decisão, linguagem estruturada, condições pré/pós, fluxogramas e outras. Qualquer uma destas especificações pode ser empregada, desde que satisfaçam dois requisitos essenciais:

- a especificação de processos deve ser expressa de uma forma que possa ser verificada pelo usuário e pelo analista de sistemas;
- a especificação de processos deve ser expressa de uma forma que possa ser efetivamente comunicada às diversas pessoas envolvidas.

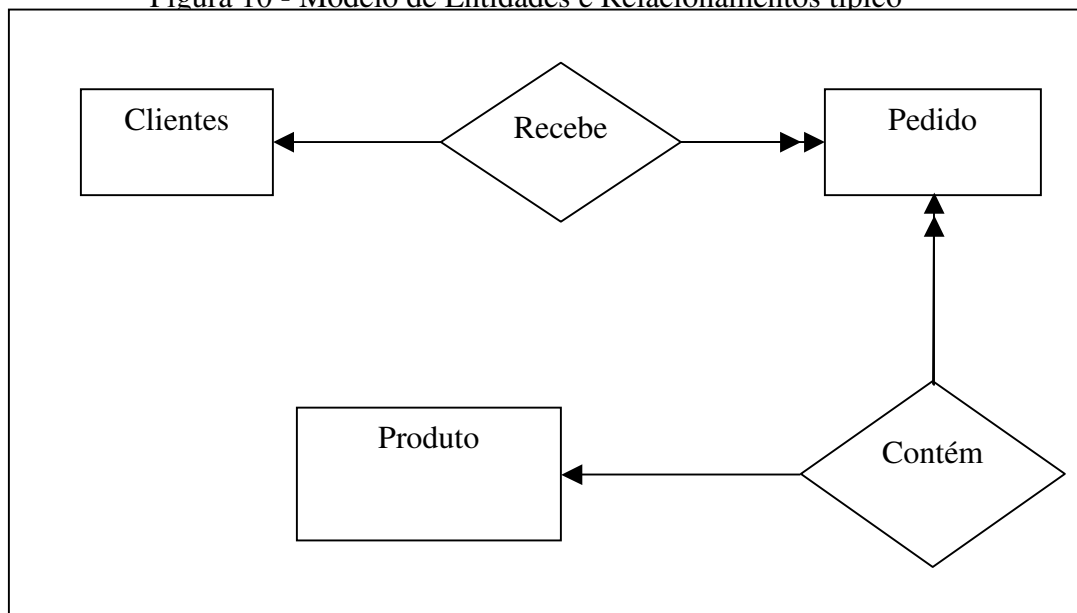
Os itens como diagrama de fluxo de dados, dicionário de dados e especificação de processos mostram o que o sistema faz, descrevendo suas funções e procedimentos. Existe ainda um recurso que descreve um modelo conceitual de dados para o sistema que é denominado de modelo de entidades e relacionamentos (MER).

### **4.3.4 MODELO DE ENTIDADES E RELACIONAMENTOS**

De acordo com Yourdon (1990), o modelo de entidades e relacionamentos pode ser definido como um modelo em rede que descreve a diagramação dos dados armazenados de um sistema em alto nível de abstração, conforme observado na Figura 10.

Esse gráfico facilita a visualização da organização e o inter-relacionamento entre os objetos que são conectados uns aos outros através de relacionamentos representando um conjunto de conexões entre objetos representados por losango.

Figura 10 - Modelo de Entidades e Relacionamentos típico



Fonte: Adaptado de Yourdon (1990)

Segundo Yourdon (1993), o diagrama de entidade e relacionamento é uma ferramenta de modelagem que é utilizada para modelar regras particulares de importância para a empresa e o relacionamento entre elas, sendo utilizado para identificar e organizar a informação utilizada pela empresa, não apenas a informação armazenada.

As entidades são os objetos definidos, podendo ser objetos específicos como pessoas ou uma nota fiscal, ou conceitos abstratos, com posições e serviços, sendo representadas por um retângulo no DER, tendo um único nome que deverá refletir o tipo do objeto (DEMARCO, 1989) e (YOURDON, 1993).

Cada ocorrência do relacionamento corresponde a uma associação de exatamente uma ocorrência de cada entidade que participa do relacionamento (DEMARCO, 1989).

Maiores detalhes sobre a análise estruturada poderão ser encontrados em: Demarco(1989), Gane (1984), Martin(1991) e Yuordon(1990).

## 4.4 POWER DESIGN

De acordo com Fischer (1990), o *Power Designer*, aceita diversos níveis de abstração do projeto. No nível mais alto, estão os diagramas de fluxo de dados, que podem “explodir”, transformando-se em outros, de níveis mais baixos, gráficos estruturais, diagramas estruturais, ou diagramas de relacionamento de entidades. O *Power Designer* é uma ferramenta *CASE*, que integra a metodologia de análise estruturada Yourdon/DeMarco á metodologia de dados e do projeto estruturado.



## 5 DESENVOLVIMENTO DO APLICATIVO

O resultado do trabalho foi a criação de um aplicativo que auxilie o agente de trânsito municipal no preenchimento dos autos de infrações, aplicando-se algumas das técnicas e teorias anteriormente vistas.

Nesta seção será descrita a especificação utilizada para o desenvolvimento do aplicativo, bem como os requisitos e técnicas para a implementação do mesmo, além de um exemplo de sua utilização.

Os softwares modernos procuram ter uma interface mais simples e amigável com os usuários proporcionando um melhor entendimento e facilidade de seu uso. Essas interfaces permitem o desenvolvedor criar visualmente a interface como o usuário utilizando-se apenas do mouse.

### 5.1 ESPECIFICAÇÃO DO APLICATIVO

O aplicativo que foi desenvolvido para a emissão dos autos de infrações, tem por objetivo auxiliar o agente de trânsito no correto preenchimento dos autos de infrações por ele emitidos, reduzir o número de anulações de autos de infrações em decorrência de erros de escrita, facilitar o processo de envio dos autos de infrações para o setor responsável pela emissão das notificações de infração, através do arquivo texto gerado pelo aplicativo.

Para a especificação do aplicativo deste Trabalho de Conclusão de Curso foi utilizada a ferramenta de programação Genexus na versão 8.0, com a geração do código na linguagem *Embedded Visual Basic*. O aplicativo foi instalado no aparelho *Pocket PC MIO 338*.

### 5.1.1 ANÁLISE DE REQUISITOS

De acordo com Tonsig (2003), a análise e especificação de requisitos de software envolve atividades que determinarão os objetivos de um software e as restrições associadas a ele. Abaixo tem-se a análise de requisitos do aplicativo:

Cadastrar Multas:

- permitir a manutenção das multas de acordo com CTB.

Cadastrar ruas:

- permitir a manutenção das ruas existentes no município.

Cadastrar agentes:

- permitir a manutenção dos agentes que farão uso do aplicativo.

Cadastrar veículos:

- permitir a manutenção dos veículos cadastrados junto ao CIASC.

Cadastrar infrações:

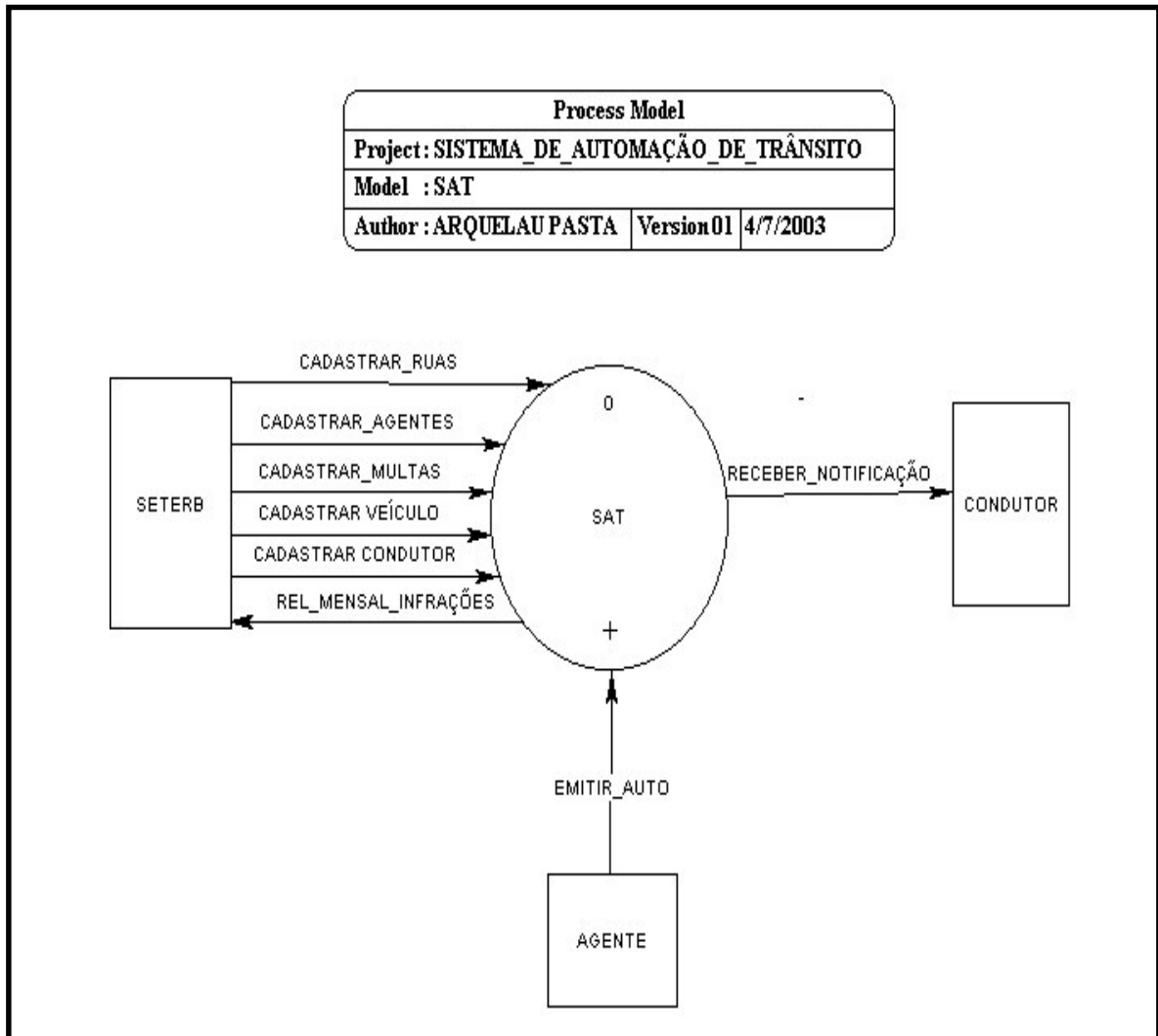
- permitir a manutenção das infrações cometidas;
- permitir a emissão do auto de infração a ser entregue ao infrator quando necessário;
- permitir consulta sobre dados referentes aos veículos autuados;
- permitir consulta sobre a rua na qual ocorreu a infração;
- permitir consulta sobre a multa na qual enquadra-se a infração cometida pelo infrator;

permitir o envio dos autos emitidos ao CIASC, através do sincronismo com o computador base que encontra-se no SETERB.

## 5.1.2 DIAGRAMA DE CONTEXTO

Na **Erro! A origem da referência não foi encontrada.** tem-se o diagrama de contexto do aplicativo, gerado pela ferramenta Power Design:

Figura 11- Diagrama de contexto do aplicativo



### 5.1.3 DICIONÁRIO DE DADOS

Abaixo visualiza-se o DD do aplicativo, gerado também pela ferramenta *Power*

*Design:*

#### AGENTE

##### Column List

| Name        | Code        | Type       | P   | M   |
|-------------|-------------|------------|-----|-----|
| AGENTE      | AGENTE      | numeric(3) | Yes | Yes |
| NOME_AGENTE | NOME_AGENTE | char(20)   | No  | Yes |

#### CONDUTOR

##### Column List

| Name        | Code        | Type        | P   | M   |
|-------------|-------------|-------------|-----|-----|
| CNH         | CNH         | numeric(11) | Yes | Yes |
| NOME        | NOME        | char(30)    | No  | Yes |
| ENDERECO    | ENDERCO     | char(20)    | No  | Yes |
| NUMERO      | NUMERO      | numeric(5)  | No  | No  |
| COMPLEMENTO | COMPLEMENTO | char(15)    | No  | No  |
| COND_MUNIC  | COND_MUNIC  | char(30)    | No  | Yes |
| EST_COND    | EST_COND    | char(2)     | No  | No  |
| CPFNPJ      | CPFNPJ      | char(14)    | No  | Yes |

#### INFRAÇÃO

##### Column List

| Name       | Code       | Type        | P   | M   |
|------------|------------|-------------|-----|-----|
| ORGAO      | ORGAO      | numeric(3)  | Yes | Yes |
| NR_AUTO    | NR_AUTO    | numeric(8)  | Yes | Yes |
| AGENTE     | AGENTE     | numeric(3)  | No  | No  |
| COD_RUA    | COD_RUA    | numeric(5)  | No  | No  |
| COD_MULTA  | COD_MULTA  | numeric(4)  | No  | No  |
| REFERE     | REFERE     | char(20)    | No  | No  |
| HORA       | HORA       | char(15)    | No  | Yes |
| VEL_PER    | VEL_PER    | numeric(3)  | No  | No  |
| VEL_AFE    | VEL_AFE    | numeric(3)  | No  | No  |
| RADAR      | RADAR      | numeric(1)  | No  | No  |
| NR_RADAR   | NR_RADAR   | numeric(5)  | No  | No  |
| OBSERVACAO | OBSERVACAO | char(40)    | No  | No  |
| CNH        | CNH        | numeric(11) | No  | No  |
| PLACA      | PLACA      | char(8)     | No  | No  |

## MULTA

### Column List

| Name       | Code       | Type       | P   | M   |
|------------|------------|------------|-----|-----|
| COD_MULTA  | COD_MULTA  | numeric(4) | Yes | Yes |
| DESCRICAO1 | DESCRICAO1 | char(30)   | No  | Yes |
| DESCRICAO2 | DESCRICAO2 | char(30)   | No  | No  |
| ARTIGO     | ARTIGO     | char(11)   | No  | Yes |
| GRAVIDADE  | GRAVIDADE  | char(10)   | No  | No  |

## RUA

### Column List

| Name     | Code     | Type       | P   | M   |
|----------|----------|------------|-----|-----|
| COD_RUA  | COD_RUA  | numeric(5) | Yes | Yes |
| DESC_RUA | DESC_RUA | char(30)   | No  | Yes |

## VEICULO

### Column List

| Name      | Code      | Type       | P   | M   |
|-----------|-----------|------------|-----|-----|
| PLACA     | PLACA     | char(8)    | Yes | Yes |
| ESTADO    | ESTADO    | char(2)    | No  | Yes |
| COD_MUNIC | COD_MUNIC | numeric(3) | No  | Yes |
| MUNICIPIO | MUNICIPIO | char(30)   | No  | No  |
| MARCAS    | MARCAS    | char(20)   | No  | Yes |
| MODELO    | MODELO    | char(20)   | No  | Yes |
| ESPECIE   | ESPECIE   | char(20)   | No  | Yes |
| COD_ESP   | COD_ESP   | numeric(4) | No  | Yes |
| CATEGORIA | CATEGORIA | char(20)   | No  | Yes |
| COD_CAT   | COD_CAT   | numeric(4) | No  | Yes |
| RENAVAM   | RENAVAM   | numeric(9) | No  | Yes |

### Reference List

| Reference Name | Reference Code | Parent   | Child    |
|----------------|----------------|----------|----------|
| LOCAL          | LOCAL          | RUA      | INFRACAO |
| PERTENCE       | PERTENCE       | MULTA    | INFRACAO |
| POSSUI         | POSSUI         | VEICULO  | INFRACAO |
| RECEBE         | RECEBE         | CONDUTOR | INFRACAO |

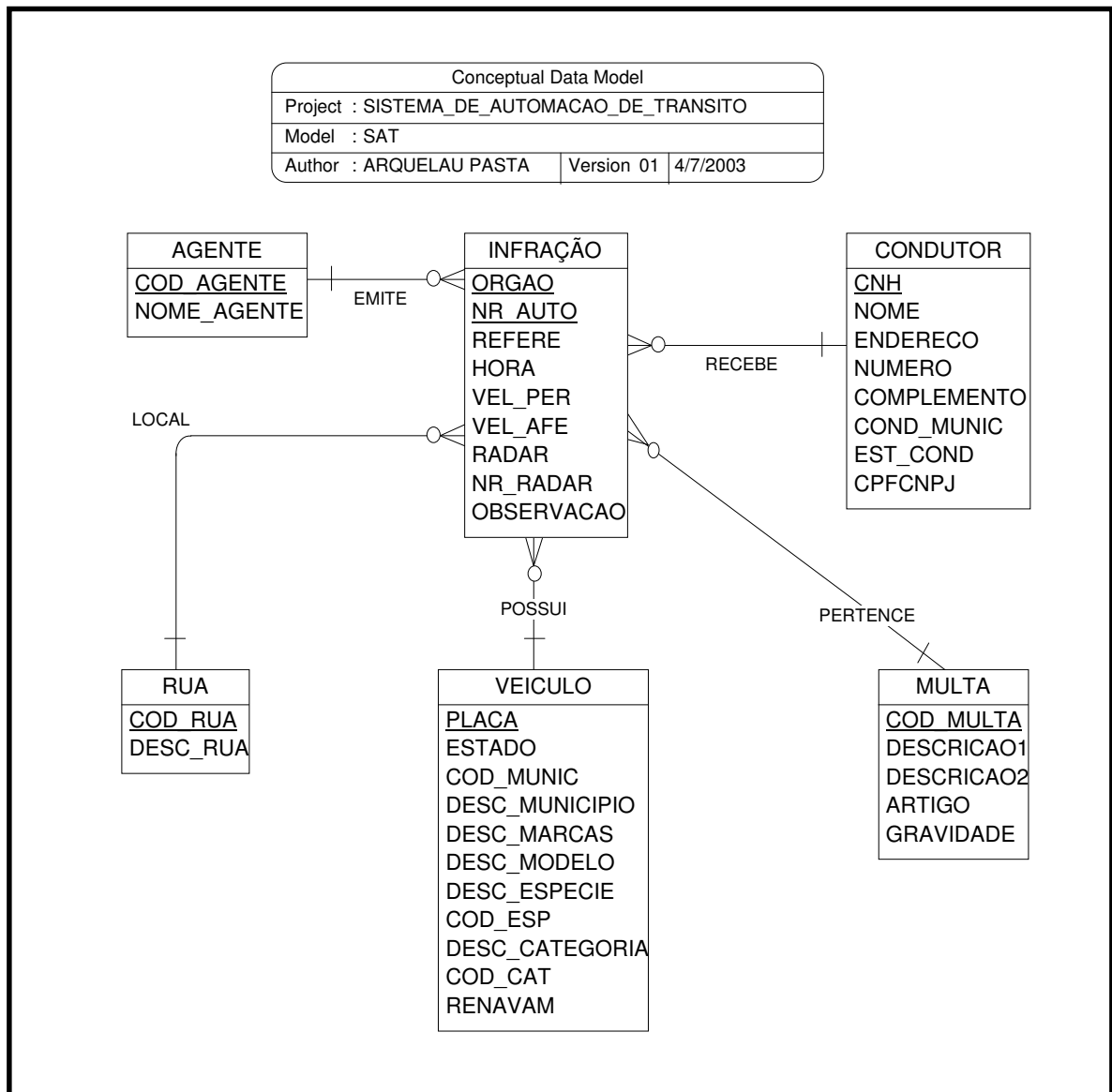
**Index List**

| <b>Table Code</b> | <b>Index Code</b> | <b>P</b> | <b>F</b> | <b>A</b> | <b>U</b> | <b>C</b> |
|-------------------|-------------------|----------|----------|----------|----------|----------|
| CONDUTOR          | CONDUTOR_PK       | Yes      | No       | No       | Yes      | No       |
| INFRACAO          | INFRACAO_PK       | Yes      | No       | No       | Yes      | No       |
|                   | RELATION_83_FK    | No       | Yes      | No       | No       | No       |
|                   | RELATION_84_FK    | No       | Yes      | No       | No       | No       |
|                   | RELATION_85_FK    | No       | Yes      | No       | No       | No       |
|                   | RELATION_86_FK    | No       | Yes      | No       | No       | No       |
| MULTA             | MULTA_PK          | Yes      | No       | No       | Yes      | No       |
| RUA               | RUA_PK            | Yes      | No       | No       | Yes      | No       |
| VEICULO           | VEICULO_PK        | Yes      | No       | No       | Yes      | No       |

### 5.1.3 DIAGRAMA DE ENTIDADE E RELACIONAMENTO

Na Figura 12 tem-se o diagrama de entidade e relacionamento do aplicativo, gerado pela ferramenta *Power Design*.

Figura 12 - Diagrama de entidade e relacionamento do aplicativo



Pode-se observar que o modelo de dados não está normatizado segundo a 3FN (terceira forma normal), uma vez que o hardware apresenta limitações em termo de estrutura armazenamento e manipulações de dados. Sugere-se para equipamentos com maior poder de armazenamento/processamento a referida normalização.

## 5.1.4 DIAGRAMA DE FLUXO DE DADOS PARTICIONADO

Na Figura 13 tem-se o DFD particionado do aplicativo:

Figura 13 - DFD particionado

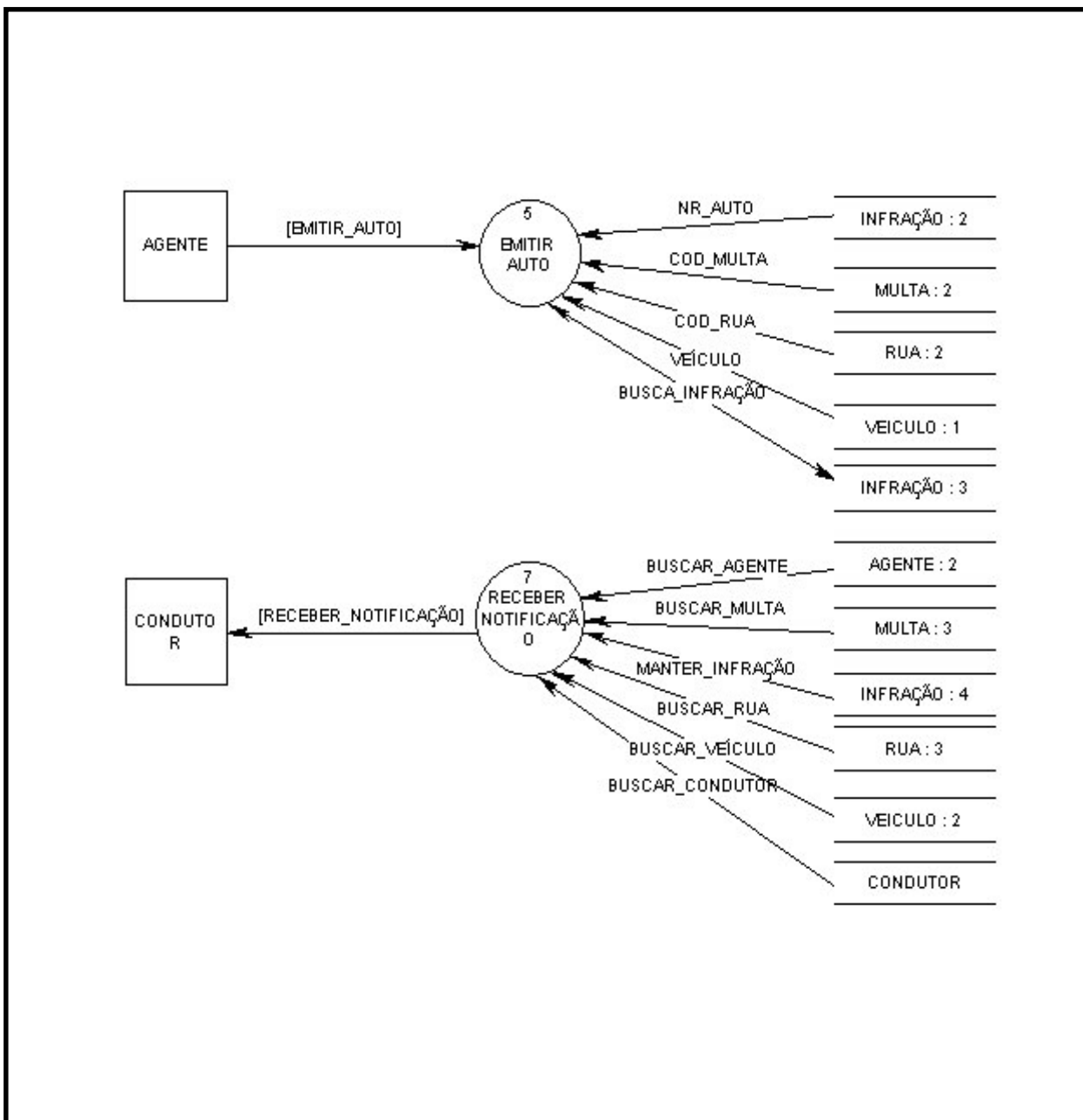
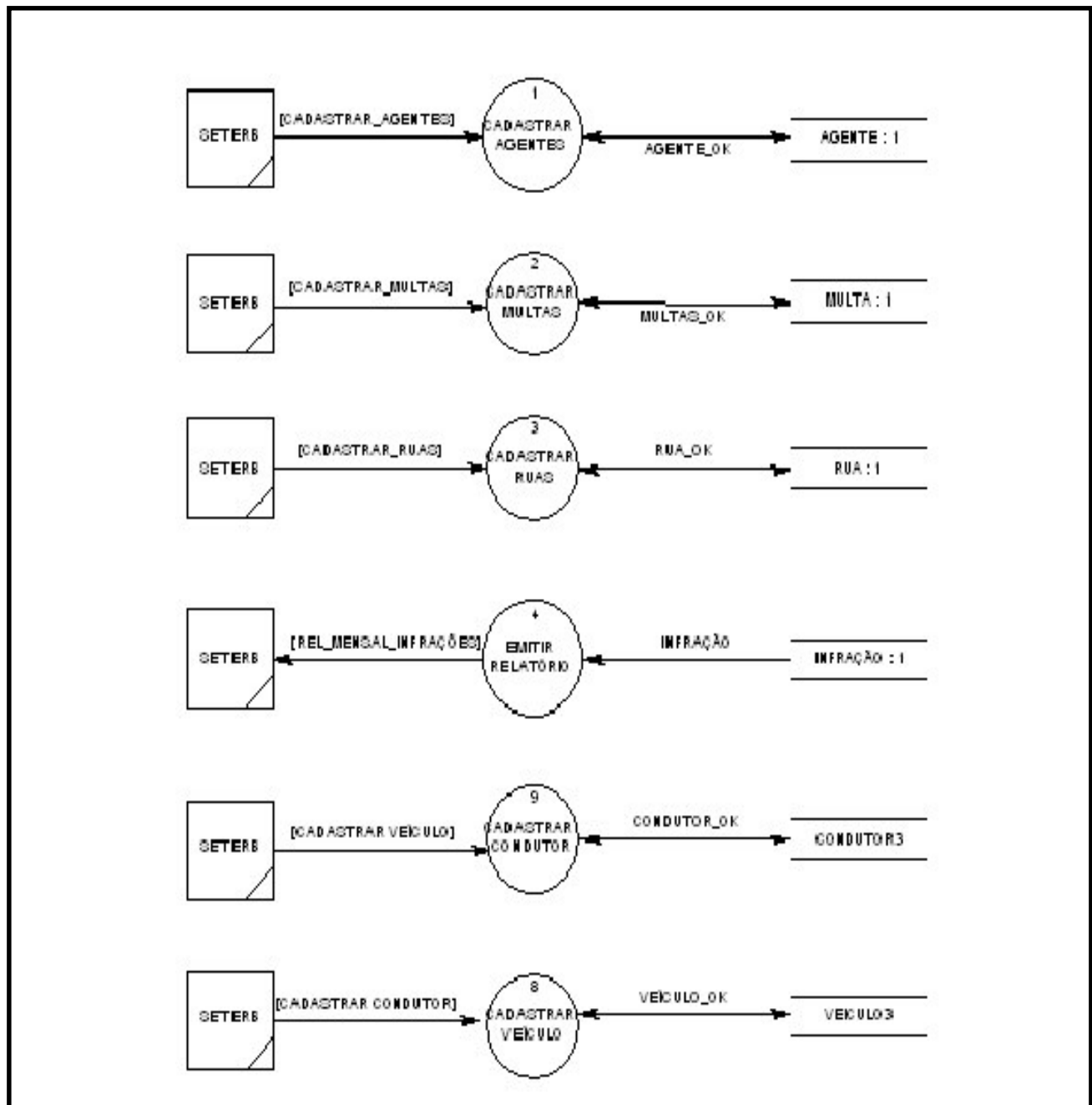




Figura 14 - Continuação do DFD particionado



### 5.1.5 MATRIZ CRUD

Uma matriz CRUD , do acrônimo em inglês (Create, Read, Update, Delete) é uma tabela que mostra as ligações entre processos e repositório de dados. Quando uma ligação existe, mostra se o processo cria, lê, atualiza ou suprime o processo.

Na Figura 15 tem-se a matriz CRUD gerada pela ferramenta *Power Design*:

Figura 15 - Matriz CRUD

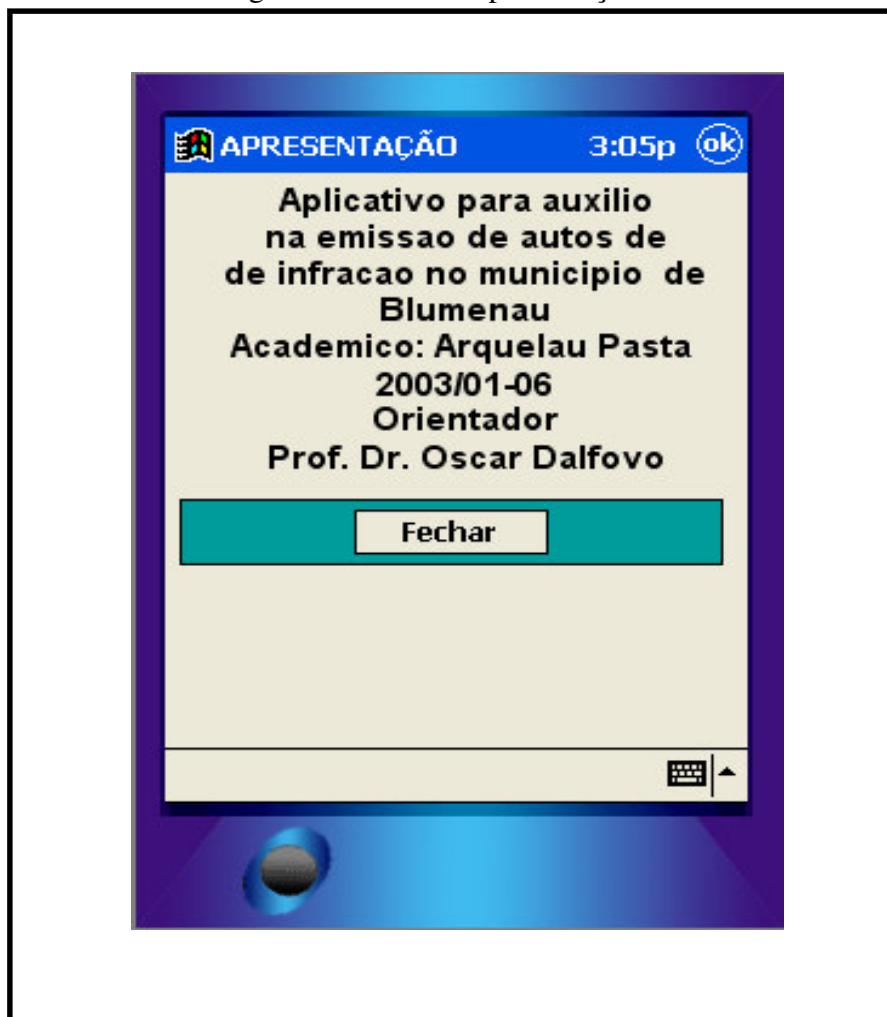
|          | CADASTRAR AGENTES | CADASTRAR CONDUTOR | CADASTRAR MULTAS | CADASTRAR RUAS | CADASTRAR VEÍCULOS | EMITIR AUTO | EMITIR | RECEBER |
|----------|-------------------|--------------------|------------------|----------------|--------------------|-------------|--------|---------|
| AGENTE   | CRUD              |                    |                  |                |                    | R           | R      | R       |
| CONDUTOR |                   | CRUD               |                  |                |                    | R           | R      | R       |
| MULTA    |                   |                    | CRUD             |                |                    | R           | R      | R       |
| RUA      |                   |                    |                  | CRUD           |                    | R           | R      | R       |
| VEÍCULO  |                   |                    |                  |                | CRUD               | R           | R      | R       |
| INFRAÇÃO |                   |                    |                  |                |                    | CRUD        | R      | R       |

## 5.2 IMPLEMENTAÇÃO DO APLICATIVO

Para implementação do aplicativo foi utilizada a ferramenta de desenvolvimento Genexus versão 8.0, gerando o código fonte na linguagem *EVV*. Cabe aqui ressaltar que esta versão da ferramenta encontra-se em fase de testes pelo fabricante Artech, desta forma tendo algumas limitações com relação as demais versões da ferramenta desenvolvida pelo fabricante.

Nesta fase elabora-se o modelo operacional do aplicativo, com a apresentação das telas de entrada de dados. Em decorrência deste trabalho, desenvolveu-se um aplicativo denominado SAT (Sistema de Automação de Trânsito). Na Figura 16 apresenta-se a tela de apresentação do aplicativo desenvolvido.

Figura 16 - Tela de apresentação



Na Figura 17 apresenta-se a tela de entrada do aplicativo. Nesta tela tem-se as opções de emissão do auto de infração, saída do aplicativo e a chamada para uma tela sobre. Esta será a única tela acessível para o agente, as demais telas de cadastro somente serão acessíveis pelos responsáveis por esta tarefa.

Figura 17 - Tela de entrada do aplicativo



Na Figura 18 apresenta-se a tela de cadastro das multas. Este cadastro será realizado pelo SETERB. As informações existentes nesta tela e seus respectivos significados são:

- Multa: código numérico das multas;
- Descrição: descrição textual das multas;
- Artigo: artigo do CTB no qual a multa está enquadrada;
- Gravidade: gravidade na qual a multa está enquadrada perante o CTB.

Figura 18 - Tela de cadastro de multas

The screenshot shows a handheld device screen with a blue header bar containing the text "Cadastro de MULTAS" and "2:16p" along with an "ok" button. The main area is a form with the following fields:

- MULTA: 5010
- DESCRIÇÃO: DIRIGIR VEIC. S/ CNH OU PERMISSÃO
- ARTIGO: 162-I
- GRAVIDADE: GRVS-3x-APP

Below the form is a navigation bar with buttons: |< > >|, OK!, SAIR, and EXCLUI. At the bottom of the screen is a full QWERTY keyboard with a numeric keypad on the left and navigation arrows on the right.

Na Figura 19 apresenta-se a tela de cadastro das ruas. Este cadastro será realizado pelo SETERB, levando em consideração as ruas e logradouros existentes no município. Os campos existentes nesta tela e seus respectivos significados são:

- Rua: código numérico das ruas;
- Nome: descrição textual do nome das ruas.

Figura 19 - Tela de cadastro de ruas



Na Figura 20 apresenta-se a tela de cadastro dos agentes. Este cadastro será inserido pelo SETERB e será baseado no quadro funcional do órgão. As informações existentes nesta tela e seus respectivos significados são:

- Agente: código numérico dos agentes;
- Nome: nome do agente.

Figura 20 - Tela de cadastro de agentes



Na Figura 21 apresenta-se a tela de cadastro dos veículos. Este cadastro será realizado pelo SETERB, com base nas informações oriundas do CIASC, órgão responsável pelo processamento de dados dos veículos cadastrados no estado de Santa Catarina. Este mesmo órgão é responsável pelo processamento os autos emitidos, gerando após o processamento a notificação a ser enviada para o proprietário do veículo autuado. As informações existentes nesta tela e seus respectivos significados são:

- Placa: placa do veículo;
- Município: descrição do município de cadastramento do veículo;
- Estado: estado de cadastramento do veículo;
- Marcas: marca do veículo;
- Modelo: modelo do veículo;
- Espécie: espécie do veículo, no qual o mesmo está enquadrado no CTB;
- Categoria: categoria do veículo, na qual o mesmo está enquadrado no CTB;
- Renavan: código da renavan destinado ao veículo.



Figura 21 - Tela de cadastro de veículos

**Cadastro de VEICULOS** 2:35p **ok**

PLACA  UF

MUNICIPIO

MARCA

MODELO

ESP  CAT

RENAVAM

|< > >| **OK!** **SAIR** **EXCLUI**

123 1 2 3 4 5 6 7 8 9 0 - = ←  
Tab q w e r t y u i o p [ ]  
CAP a s d f g h j k l ; '  
Shift z x c v b n m , . / ←  
Ctl á ü ` \ ↓ ↑ ← →

Na Figura 22 apresenta-se a tela de cadastro das infrações. Esta tela será utilizada pelo agente de trânsito para a emissão do auto de infração. Esta é a tela principal do aplicativo, através desta tela serão cadastrados os dados pertinentes a infração cometida pelo condutor. As informações existentes nesta tela e seus respectivos significados são:

- Órgão: numeração do órgão responsável pelo setor de trânsito no município;
- Nr\_Auto: numeração do auto de infração, esta numeração é fornecida pelo DETRAN;
- Placa: placa do veículo autuado;
- Renavam: Código do Renavam do veículo autuado;
- Rua: endereço da autuação;
- Referência: ponto de referência para localização;
- Data: data da autuação;
- Multa: código da multa, de acordo enquadramento no CTB;
- Per.: velocidade permitida para o local;
- Afe.: velocidade aferida, quando da infração;
- Radar: indicativo de utilização de equipamento de radar;
- Nr. Radar: número do radar;
- CNH: Carteira Nacional de Habilitação do condutor autuado;
- Observação: texto descritivo;
- Agente: código do agente emissor do auto da infração.

Figura 22 - Tela de cadastro de infrações

**INFRACAO** 3:19p **ok**

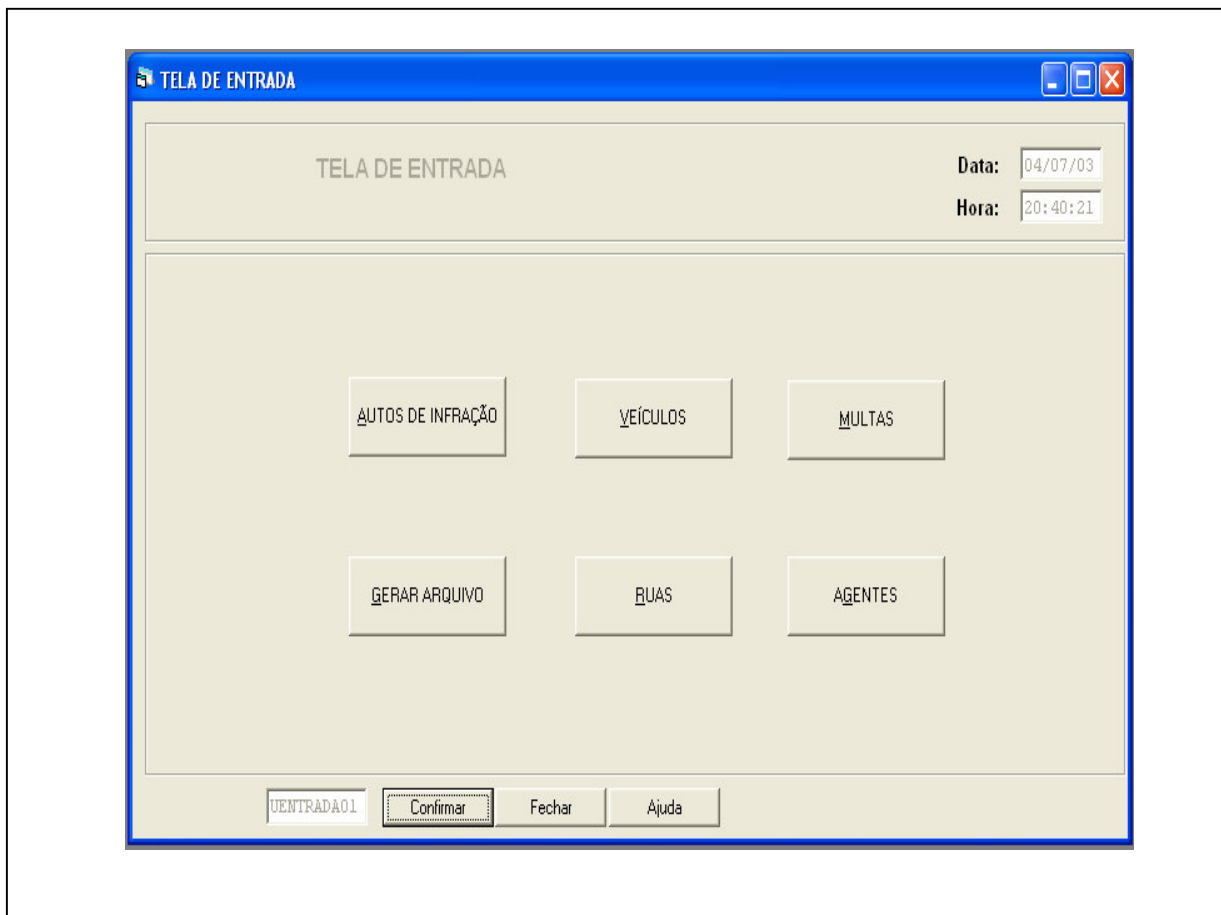
|            |                |         |                        |
|------------|----------------|---------|------------------------|
| ORGÃO      | 34             | NR.AUTO | 55525641               |
| PLACA      | LZT-9865       | RENAVAM | 700922296              |
| RUA        | 1              | ?       | B RAZ WANKA            |
| REFERÊNCIA | CAMPUS IV      | DATA    | 10/6/03                |
| MULTA      | 5010           | ?       | DIRIGIR VEIC. S/CNH OU |
| RADAR      | NÃO            | 0       | PERM. 0 AFER. 0        |
| CNH        | 117943412      | OBS     | SEM USO DE LENTES      |
| CONDUTOR   | ARQUELAU PASTA |         |                        |
| AGENTE     | 1              | ?       | EVERALDO               |

|< > >| **OK!** **SAIR**

⌨ ▲

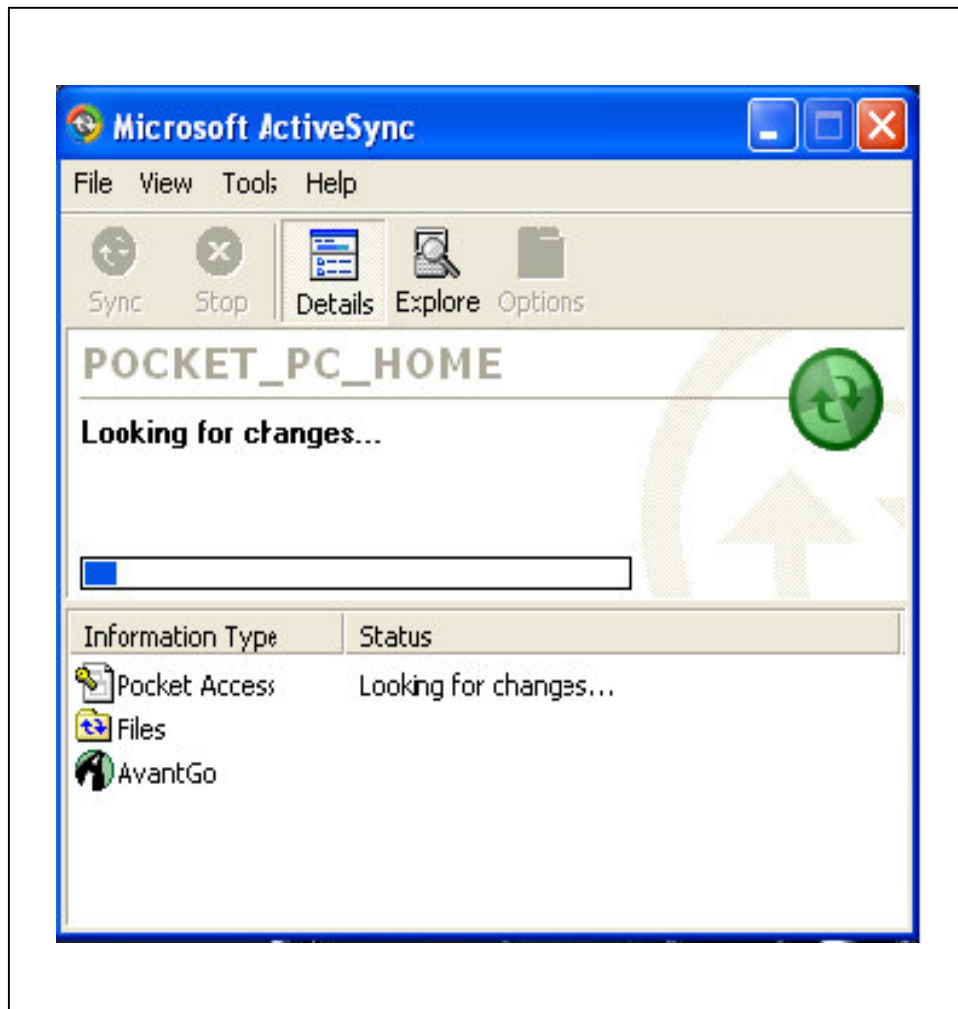
Na Figura 23 tem-se a tela do aplicativo instalado no SETERB, será nesta tela que far-se-á a geração do arquivo texto, em seguida o mesmo será encaminhado ao órgão responsável pela emissão da notificação de infração.

Figura 23 - Tela do aplicativo no SETERB



Na Figura 24, tem-se a tela de sincronismo entre o *palmtop* e o PC, é neste momento que é feito a atualização dos dados para o futuro envio ao CIASC.

Figura 24 - Tela de sincronismo de dados



## 6 CONCLUSÃO

O tema sobre PDA é uma área recente que está em evolução contínua e portanto é um tema pouco difundido no meio acadêmico e profissional. Devido a pouca difusão do tema encontrou-se dificuldade no levantamento de material referente ao assunto proposto neste trabalho. Verifica-se uma grande deficiência com relação às linguagens de programação para este tipo de computador. Um fator importante a ser considerado para programação para PDA ou *hand-helds*, hoje, são suas limitações quanto a memória, o sistema operacional de cada equipamento, mas isto é algo que está sendo superado tanto pelos fabricantes deste tipo de equipamento, quanto pelas empresas que desenvolvem linguagens de programação para os mesmos.

A automação de serviços é um aspecto que está em crescente desenvolvimento. Nos serviços públicos, como o SETERB, por exemplo, a automação vem para facilitar e desburocratizar os serviços prestados, trazendo para os munícipes uma qualidade maior nos serviços prestados.

Para a implementação deste protótipo utilizando-se da ferramenta de programação Genexus que oferece recursos e suporte à aplicação deste gênero, fornecendo uma interface de fácil interpretação para o usuário, porém cabe aqui ressaltar que para a implementação do aplicativo utilizou-se de uma versão que encontra-se em teste pelo fabricante, o que por sua vez acarretou em algumas dificuldades para implementação do mesmo.

Dadas às características atuais dos PDAs, em especial a sua aplicabilidade na automação de serviços públicos, conclui-se que a automação destes serviços pode ser atingida de maneira satisfatória, o que como citado anteriormente trará como um dos benefícios a melhor qualidade dos serviços prestados.

Sob o ponto de vista mercadológico, o aplicativo mostrou-se bem aceito, sendo que o mesmo encontra-se em testes juntamente com o SETERB, para sua validação e reconhecimento com meio de substituição do talonário de emissão dos autos de infração.

O objetivo principal do trabalho proposto, de desenvolver um aplicativo para o auxílio do agente de trânsito no preenchimento do auto de infração de trânsito, foi atingido.

Passo seguinte ao mesmo é a geração de um arquivo no formato texto, que será enviado ao órgão responsável pela emissão das notificações de autuação.

Em suma a elaboração do trabalho proposto foi de grande valia, pois possibilitou o conhecimento na área de programação para PDAs ou *hand-helds*, trazendo suas características e limitações. Trouxe também um aperfeiçoamento no uso da ferramenta de programação Genexus e aplicação dos conhecimentos relacionados a análise de programação ministrados durante o curso de graduação.

## **6.1 DIFICULDADES ENCONTRADAS**

Durante a elaboração deste trabalho encontrou-se dificuldade no levantamento bibliográfico e pesquisa de material referente a PDAs, sendo somente encontrado através de listas de discussões e páginas na internet.

Outra dificuldade encontrada foi em relação a versão da ferramenta de programação Genexus. Conforme citado anteriormente a ferramenta de programação Genexus, na versão 8.0, especificamente para a geração do código fonte em EVB, encontra-se em teste, assim sendo algumas funções não estão disponíveis para uso.

## **6.2 SUGESTÕES PARA TRABALHOS FUTUROS**

Como complemento a este trabalho, sugere-se a continuidade do aplicativo, visando uma melhor impressão do auto de infração.

Vislumbra-se a possibilidade de envio dos dados coletados através da internet, que poderá ser agregado ao PDA, sendo que para isto já existem aparelhos com placa de fax modem embutidas.

## REFERÊNCIAS BIBLIOGRÁFICAS

ARTECH, Advanced Reserach & Tecnology. **Vision general**. Uruguai. 1992.

BILLIAN, Otto. **Domine o trânsito uma ação mundial**. São Paulo, Distribuidora Abril S.A., 1975.

BIZZOTTO, Carlos E. Negrão, et al, **Informática básica: passo a passo, conciso e objetivo**. Florianópolis: Visual Books, 1998.

DEMARCO, Tom **Análise estruturada e especificação de sistemas**. Rio de Janeiro: Campus 1989.

DENATRAN. **Legislação**, Brasil,[1998?]. Disponível em: < <http://www.denatran.gov.br/>>. Acesso em: 01 mar. 2003.

DIAS, Paulo Roberto. **Sistema de informação baseado em regras de negócio utilizando a ferramenta genexus estudo de caso no setor têxtil**. 2002.103 f. Dissertação (Mestrado em Engenharia da Produção) – Programa de Pós-Graduação em Engenharia da Produção, Universidade Federal de Santa Catarina.

FISCHER, Alan S. **Case utilização de ferramentas para desenvolvimento de software**. Rio de Janeiro : Campus, 1990.

GANE, Chris; Sarson, Trish. **Análise estruturada de sistemas**. Rio de Janeiro: LTC-Livros Técnicos e Científicos, 1984.

JUNIOR, José Jair Dill. **Protótipo de um sistema de informações estratégicas para consultórios médicos utilizando Genexus**. 2000. 54 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

MARTIN, James. **Técnicas estruturadas e case**. Tradução Lúcia Flavia Silva. São Paulo: Makron, McGraw-Hill, 1991.

NATALE, Ferdinando. **Automação industrial**. São Paulo: Érica, 2002.

PALMLBR. **Artigos e tutoriais**, São Paulo, [2003?]. Disponível em: <<http://www.palmlbr.com.br/artigos/>>. Acesso em: 01 mar. 2003.

PALMLAND. **Handhelds**, São Paulo, [2003?]. Disponível em <<http://www.palmland.com.br/handhelds/>>. Acesso em: 01 mar. 2003.

PALMLAND. **Sistemas operacionais**, São Paulo, [2003?]. Disponível em <<http://www.palmland.com.br/wince/>>. Acesso em: 01 mar. 2003.

PRESSMANN, Roger S. **Engenharia de software**. São Paulo: Makron Books, 1995.

SETERB. **Trânsito**, Blumenau, [2003?]. Disponível em: <[http:// www.seterb.com.br/transito05.htm](http://www.seterb.com.br/transito05.htm)>. Acesso em: 01 mar 2003.




SETERB. **Coletanea da nova legislacao de transito :sinalizacao,Codigo de Transito, Resolucoes - 01 a 84/98**, codificacao das infracoes, estrutura do transito em Blumenau. - 2. ed. - Blumenau : SETERB, 1998. - 352p. :il.

TONSIG, Sérgio Luiz. **Engenharia de software**, análise e projeto de sistemas. – São Paulo: Futura, 2003.

YOURDON, Edward. **Análise estruturada moderna**. Rio de Janeiro: Campus, 1990.

YOURDON, Inc. **Yourdon systems method**. New Jersey: Prentice Hall,1993.

## ANEXO I

|  |   |  |  |
|--|---|--|--|
|  <b>ESTADO DE SANTA CATARINA</b><br><b>PREFEITURA MUNICIPAL BLUMENAU</b><br>SETERB<br>DEPARTAMENTO DE TRÂNSITO<br><b>AUTO DE INFRAÇÃO DE TRÂNSITO</b> |   | <b>1 IDENTIFICAÇÃO DO DOCUMENTO</b><br>1 - ÓRGÃO    2 - Nº DO AUTO<br><b>034    551.77967    A</b> |  |
| <b>2 IDENTIFICAÇÃO DO VEÍCULO</b>  |   |  |  |
| 1 - PLACA  |   | 2 - MUNICÍPIO  | 3 - CÓDIGO   |
| 204-7420 SC  |   | Blum   | Blum   |
| 4 - MARCA / MODELO   |   | 5 - Cód.   | 6 - ESPÉCIE / TIPO   |
| FIAT Uno   |   | 200  | Pass. Auto   |
| 8 - CATEGORIA  |   | 9 - Cód.   | 10 - RENAVAM   |
| part   |   | 644811765  |  |
| <b>3 INFRAÇÃO</b>  |   |  |  |
| 1 - LOCAL DA INFRAÇÃO  |   | 2 - Nº / REFERÊNCIA  |  |
| Avenida São Paulo  |   | 320  |  |
| 3 - DATA   | 4 - HORA  | 5 - MUNICÍPIO  | 6 - CÓDIGO   |
| 10-02-2002   | 03:25   | BLUMENAU   | 80470  |
| <b>4 ENQUADRAMENTO DA INFRAÇÃO</b>   |   |  |  |
| 545 2  | Estacionamento na calçada, faixa pedestres, etc.        | 555 0  | Estac. local proibido - placa proibido estacionar R-6A       |
| 548 7  | Estacionar em fila dupla                                | 556 8  | Estac. em local proibido - placa proibido parar / estacionar |
| 550 9  | Estacionar ponto de embarque desembarque de ônibus      | 574 6  | Trânsito local / horário não permitido para todos veículos   |
| 552 5  | Estacionar na contramão de direção                      | 583 5  | Desobedecer ordens autoridade trânsito ou agentes            |
| 554 1  | Estac. desac. com regulamento Sinalização (Estac. Reg.) | 599 1  | Retorno em local proibido                                    |
| 681 5  | TRANSITAR PRODUZINDO FUMAÇA ACIMA DO PERMITIDO          | Nível  | Porcentagem  |
| CÓDIGO   | VEL. PERMITIDA Km/h                                     | VEL. AFERIDA Km/h  | RADAR Nº   |
| 5055   | TRANSITAR EM VELOCIDADE SUPERIOR PARA O LOCAL           |  |  |
| CÓDIGO   | DESCRIÇÃO DA INFRAÇÃO                                   |  |  |
| 5055   | Dirigir Veic s/ o uso de lentos corretivos              |  |  |
| <b>5 IDENTIFICAÇÃO DO CONDUTOR</b>   |   |  |  |
| 1 - NOME   |   |  |  |
| Eduardo A. Volles  |   |  |  |
| 2 - ENDEREÇO   |   | 3 - NÚMERO   | 4 - COMPLEMENTO  |
| Tocantins  |   | 210  | Centro   |
| 5 - MUNICÍPIO  |   | 6 - UF   | 7 - CEP  |
| Blum   |   | SC   |  |
| 8 - Nº REGISTRO CNH  |   | 9 - UF   | 10 - CPF / CNPJ / IDENT.                                     |
| 01214197053  |   | SC   | 3666415  |
| 11 - ASSINATURA  |   |  |  |
| Eduardo A. Volles  |   |  |  |
| <b>6 IDENTIFICAÇÃO DO INFRATOR</b>   |   |  |  |
| 1 - NOME   |   | 2 - CPF / CNPJ / IDENT.  |  |
| ACIMA  |   |  |  |
| <b>7 OBSERVAÇÃO</b>  |   |  |  |
| s/ lentos corretivos   |   |  |  |
| <b>8 IDENTIFICAÇÃO DO AGENTE</b>   |   |  |  |
| 1 - AGENTE Nº  | 2 - NOME AGENTE   |  | 3 - ASSINATURA   |
| 390  | Archeide  |  | Archeide   |

1ª VIA - Digitação - 2ª VIA - Infrator - 3ª VIA - Arquivo