

6,90

(seis e nove)

JOSÉ EDUARDO FREIRE DAMIÃO
MARCELO SHOITI GUNJI

SISTEMA DE FISCALIZAÇÃO DE VEÍCULOS

Trabalho de Formatura apresentado ao
Departamento de Engenharia Mecânica da
Escola Politécnica da Universidade de São
Paulo para a obtenção do diploma de
graduação.

São Paulo
2002

**JOSÉ EDUARDO FREIRE DAMIÃO
MARCELO SHOITI GUNJI**

SISTEMA DE FISCALIZAÇÃO DE VEÍCULOS

Trabalho de Formatura apresentado ao
Departamento de Engenharia Mecânica da
Escola Politécnica da Universidade de São
Paulo para a obtenção do diploma de
graduação.

Orientador:
Prof. Doutor João Paulo Marcicano.

**São Paulo
2002**

Aos meus pais e familiares, por todo o
apoio e ajuda durante esta jornada.

José Eduardo

Dedico este trabalho a todos que contribuíram diretamente ou indiretamente para minha formação. Aos meus pais e familiares, pela educação, apoio e todo suporte proporcionado ao longo de minha vida. À Susana, minha namorada, pelo incentivo, compreensão e paciência ao longo dos anos. Ao São Longuinho, pela ajuda na superação de grandes desafios.

Marcelo Shoiti Gunji

RESUMO

O modelo atual do sistema de fiscalização de veículos no Brasil é muito ineficiente, fato que incentiva os altos índices de roubo de veículos. Outra consequência é a grande quantidade de veículos em circulação com documentação irregular.

Este trabalho tem como objetivo propor um sistema de fiscalização de veículos otimizado, de forma a melhorar a situação atual. Neste trabalho é apresentada a descrição do funcionamento do sistema proposto, a escolha do sistema de aquisição; empregando o método da Matriz de Decisão; o dimensionamento e especificação do sistema proposto e a implementação do protótipo do sistema de banco de dados central.

SUMÁRIO

LISTA DE TABELAS

LISTA DE FIGURAS

1. Introdução.....	1
2. Descrição do Sistema	4
3. Descrição do Funcionamento	7
4. Descrição dos Módulos do Sistema de Fiscalização de Veículos	9
4.1. Módulo de Captura de Identificação do Veículo	9
4.1.1. Identificação por Transponder.....	9
4.1.2. Identificação por Processamento de Imagem.....	11
4.2. Sistema de Transmissão.....	12
4.3. Central de Dados.....	13
4.4. Módulo de Exibição de Status de um Veículo.....	14
5. Seleção do Sistema de Aquisição da Identificação.....	16
5.1. O Método Escolhido.....	16
5.2. Critérios de Seleção.....	17
5.3. Definição da Matriz de Decisão.....	19
5.4. Matriz de Decisão.....	24
6. Requisitos do Sistema	25
7. Projeto do Módulo de Captura de Identificação do veículo	26
7.1. Introdução	26
7.2. Ambiente de Operação	26
7.2.1. Faixa de Temperatura de Operação	27
7.2.2. Umidade Relativa	28
7.3. Tranponder	28
7.3.1. Codificação do Transponder	29
7.4. Sistema de Leitura	31
7.4.1. Sistema de Alimentação.....	31
7.4.2. Emissor de sinais microondas e antena receptora.....	32
7.4.3. Sistema de Processamento Local.....	37

7.4.4.	Interconexão do Sistema de Processamento Local e Antena	39
7.5.	Sistema de Processamento Central	39
8.	Dimensionamento do Sistema de Transmissão	43
8.1.	Premissas	43
8.2.	Sistema de transmissão: Sistema de aquisição de dados - Central de dados ..	44
8.3.	Sistema de Transmissão: Central de dados – Ponto de fiscalização.	45
9.	Sistema de Transmissão	47
9.1.	Introdução	47
9.2.	O Sistema de Rádio	47
9.2.1.	Especificações Técnicas – Sistema de Rádio	49
9.3.	Linha Privada – 256 Kb/s	51
10.	Plano de Projeto de Software	52
10.1.	Descrição Rápida do Funcionamento do Software	52
10.2.	Requisitos de Projeto de Software	52
10.2.1.	Introdução.....	52
10.2.2.	Marcas.....	52
10.2.3.	Escopo.....	52
10.2.4.	Conceitos, terminologia e abreviaturas.....	52
10.2.5.	Descrição Geral	53
10.3.	Modelagem do Banco de Dados e Tabelas Access:	57
10.3.1.	Tabela de Informações sobre o Veículo e sua Documentação (veículos)57	
10.3.2.	Tabela de Informações sobre os Pontos de Fiscalização de Veículos (locais) 58	
10.4.	Transações:	59
10.5.	Demonstração do funcionamento do aplicativo do sistema de fiscalização de veículos 64	
10.5.1.	Atualizar o Banco de Dados.....	65
10.5.2.	Inserir Veículo e Inserir Local.....	65
10.5.3.	Consultar Veículo e Consultar Local.....	67
10.5.4.	Resultado de Consulta de Veículo ou de Local de Fiscalização de Veículos Cadastrado.....	68

10.5.5. Alteração de Registro de Veículo e Local de Fiscalização de Veículos do Banco de Dados.....	69
10.5.6. Efetuar Controle	70
11. Módulo de Exibição de Veículos Irregulares	72
Níveis de Proteção.....	121
Bibliografia	122

APÊNDICE A

APÊNDICE B

BIBLIOGRAFIA

LISTA DE FIGURAS

Figura 1: Número de veículos roubados e furtados por ano.....	2
Figura 2: Sistema típico de fiscalização policial.	3
Figura 3: Sistema de rastreamento de veículos.	5
Figura 4: Fluxograma do sistema proposto.	7
Figura 5: Sistema de fiscalização proposto.	8
Figura 6: Sistema de transponder.....	10
Figura 7: Configuração típica do sistema de leitura de placas de veículos.....	12
Figura 8: Estruturação do sistema de banco de dados.	14
Figura 9: Evolução das temperaturas médias e mínimas anuais	27
Figura 10: Princípio de funcionamento do tag.	29
Figura 11: Projeto da alimentação do sistema de aquisição de dados.	32
Figura 12: Funcionamento da antena.	33
Figura 13: Captação da identificação do veículo pela antena.	34
Figura 14: Directividade da antena.	34
Figura 15: Antena posicionada acima dos veículos I.....	35
Figura 16: Antena posicionada acima dos veículos II.	35
Figura 17: Antena posicionada lateralmente I.....	36
Figura 18: Antena posicionada lateralmente II.....	36
Figura 19: Antena posicionada lateralmente para múltipla faixas.....	37
Figura 20: Arquitetura do sistema de aquisição de dados.....	38
Figura 21: Conexão entre o sistema de processamento local e a antena.....	39
Figura 22: Sistema de processamento de dados central	40
Figura 23: Configuração ponto-multiponto - RS 485.....	41
Figura 24: Configuração ponto-multiponto - RS 422.....	41
Figura 25: Configuração ponto-a-ponto – RS232, RS 422, RS-485.	41
Figura 26: Fluxograma do sistema de fiscalização automática de veículos.....	44
Figura 27: Diagrama em blocos do sistema de transmissão.....	47
Figura 28: Topologia típica de uma rede banda larga sem fio.	48
Figura 29: Ligação entre o sistema de aquisição de dados e o subscriber unit.	48
Figura 30: Sistema de transmissão de dados via rádio.....	49

Figura 31: Esquema de Funcionamento do Programa do S.F.V.	60
Figura 32: Menu Inicial.	64
Figura 33: Menu Atualizar Banco de Dados.	65
Figura 34: Tela de Inserção de Veículos.	66
Figura 35: Tela de Operação Concluída.	67
Figura 36: Tela de Consulta de Veículos.	68
Figura 37: Tela de Resultado de Consulta de Registro de Veículo.	69
Figura 38: Tela de Alteração de Registro de Veículo.	70
Figura 39: Tela de Efetuação de Controle Sobre uma Via.	71
Figura 40: Sistema de exibição dos dados dos veículos irregulares.	72

LISTA DE TABELAS

Tabela 1: Critérios de Seleção.....	18
Tabela 2: Matriz de Decisão.....	24
Tabela 3: Tipo de cabo X Máxima distância de transmissão	39
Tabela 4: Termologia Utilizada no Plano de Projeto de Software	53
Tabela 5: veículos: Tabela de Informações sobre o Veículo e sua Documentação....	57
Tabela 6: locais: Tabela de Informações sobre os Pontos de Fiscalização de Veículos	58

1. INTRODUÇÃO

Nos anos recentes, a sociedade tem se tornado mais sensível com relação à ocorrência de crimes e com à segurança em locais públicos e parece estar aceitando mais facilmente sistemas de segurança nestes locais, ainda que isto possa custar a perda de liberdades individuais.

Em especial, existe uma preocupação por parte da sociedade com relação ao roubo e furto de veículos e com como a polícia pode realizar um controle mais eficaz sobre a circulação dos mesmos para tentar inibir a ocorrência destes tipos de crimes. As estatísticas mostram que os crimes relacionados a furto e roubo de automóveis vêm crescendo ano a ano juntamente com o aumento populacional das principais cidades do Brasil. Além disto, o número de ocorrências de roubo de cargas também aumentou significativamente na última década implicando em grande prejuízo não só para a sociedade civil como para empresas privadas tais quais transportadoras e seguradoras.

Existem sistemas de controle por parte da polícia tanto militar como rodoviária federal para tentar conter a ocorrência destes tipos de crimes ou solucioná-los, como, por exemplo, os postos de controle da polícia rodoviária federal, existentes nas principais rodovias do país, e os eventuais postos de comando da polícia militar, que param motoristas nas cidades e estradas em buscas de irregularidades nos veículos.

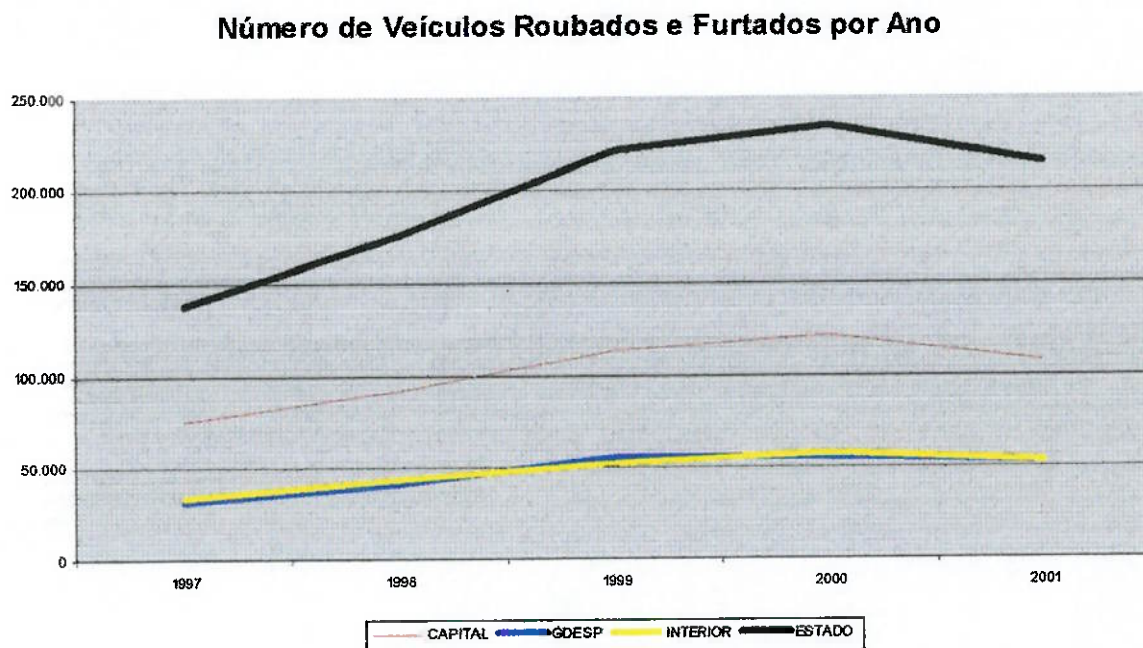


Figura 1: Número de veículos roubados e furtados por ano.

Este tipo de controle não é totalmente eficaz, pois depende da capacidade do policial de parar e revistar os carros, o que torna-se muito complicado em vias de intensa circulação de veículos, logo, nem todos os motoristas são parados para ter os seus documentos checados, permitindo-se a passagem de veículos irregulares sem que nenhuma providência com relação aos mesmos seja tomada.

Além disto, a parada ou não de um motorista depende da escolha do policial rodoviário que esta executando esta tarefa e de critérios subjetivos e pessoais deste policial, o que ocasiona mais um ponto em que o controle pode falhar, pois na maioria dos casos não se sabe a priori quais os veículos que estão em situação irregular apenas olhando-se para eles.

Como se isto tudo não fosse o bastante, o sistema atual de fiscalização é feito com a interrupção parcial do tráfego, podendo gerar congestionamentos. Com isto, o condutor do veículo em situação irregular possui a chance de, ao avistar a fiscalização, mudar de rota pegando um caminho alternativo, evitando assim a fiscalização.

Na figura abaixo, podemos verificar como funciona um sistema típico de fiscalização:

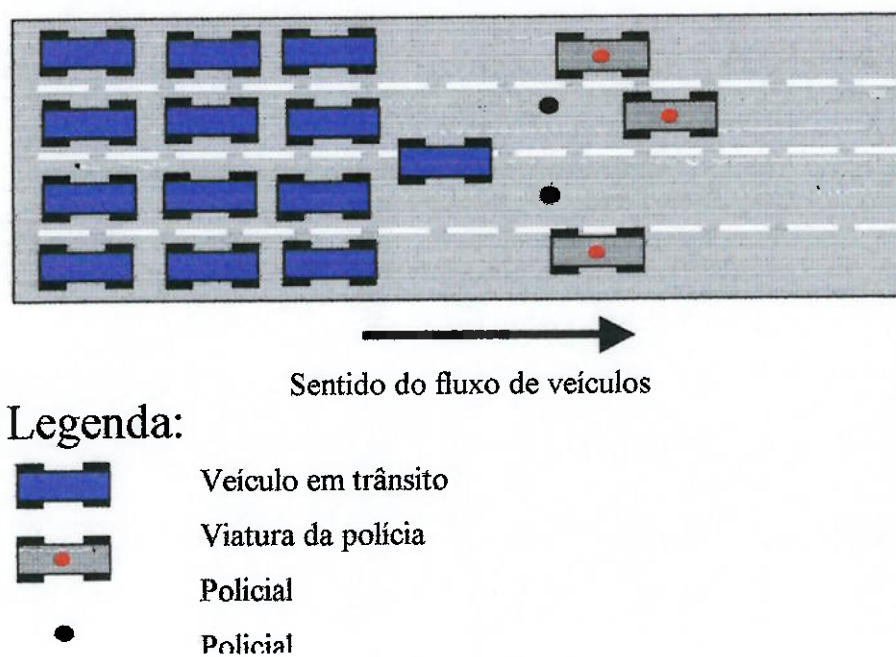


Figura 2: Sistema típico de fiscalização policial.

2. DESCRIÇÃO DO SISTEMA

Para a satisfação dos objetivos propostos acima proporemos um sistema capaz de realizar a fiscalização automática dos veículos circulantes em uma via. O sistema deverá ser capaz de fiscalizar as faixas de uma determinada via, para os carros que passarem abaixo de uma determinada velocidade especificada pelo sistema.

O sistema deverá ser alimentado com ocorrências de furto e roubo de veículos e demais irregularidades relacionadas ao tráfego de veículos de todo o território nacional. Os DETRANS estaduais já possuem bancos de dados com informações das procedências dos veículos que podem ser usados para alimentação do sistema e são os responsáveis pela regularização dos veículos de circulação em território nacional. Desta forma, o sistema deverá identificar veículos que constem como roubados ou furtados e registrar a passagem destes veículos pelo ponto de fiscalização automática e, o sistema deverá informar a ocorrência de passagem de veículo roubado ou furtado para o posto de fiscalização policial mais próximo ou à frente, no sentido da via.

O sistema deverá, também, identificar outros tipos de veículos que possuam restrições de circulação judiciais, como, por exemplo: ausência de licenciamento, ausência de pagamento de seguro obrigatório, ausência de pagamento de IPVA, etc e, o sistema deverá informar a ocorrência de passagem de veículo roubado ou furtado para o posto de fiscalização policial mais próximo ou à frente, no sentido da via.

Se por acaso o sistema identificar a passagem de um veículo qualquer e, no entanto, não puder realizar a fiscalização deste, poderá informar a ocorrência de erro de fiscalização não realizada para o posto de fiscalização policial mais próximo ou à frente, no sentido da via.

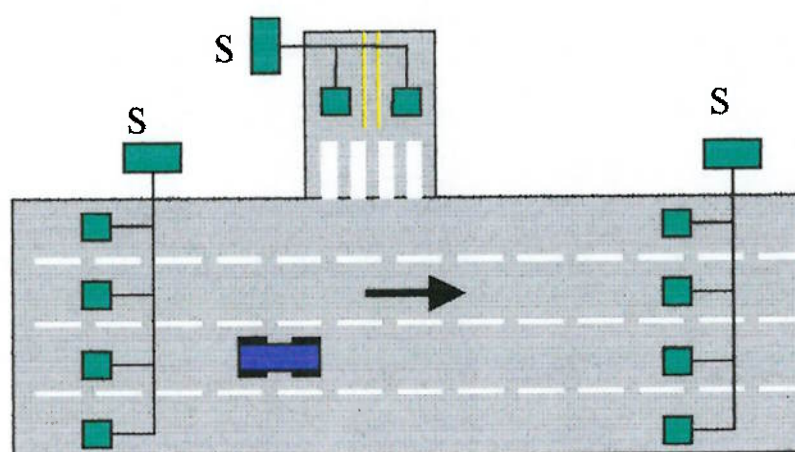
Adicionalmente a função de fiscalização de veículos, este sistema poderá exercer as seguintes funções:

- Rastreamento de veículos.
- Aplicação de multa (Excesso de velocidade, rodizio, semáforo etc.).

Para a implementação da função de rastreamento de veículos, a precisão da posição onde o veículo se encontra, dependerá do número de pontos de controle implementados e da logística empregada para relacionar estes pontos de controle.

Como exemplo, veja a figura 3, ilustrada abaixo, suponha que o veículo passou pelo sistema de identificação 1 (S_1) e não passou em S_2 e S_3 . Podemos concluir então que o veículo se encontra em algum lugar entre os sistemas de identificação 1,2 e 3.

Já para o sistema de multa por excesso de velocidade ou passagem em farol vermelho, podemos utilizar o sistema de fiscalização proposto juntamente com o sistema de detecção de velocidade ou detecção de passagem no farol vermelho. Atualmente, o veículo que cometeu a infração é fotografado, e a identificação é feita posteriormente, de forma visual. Com o sistema proposto, o veículo seria identificado eletronicamente, e a multa poderia ser emitida automaticamente.



Legenda

-  Veículo em trânsito
-  Sistema de identificação do veículo

Figura 3: Sistema de rastreamento de veículos.

No caso de multa por circulação em área e horário não permitidos (rodízio de veículos), o sistema de processamento de dados deve verificar o dia da semana, o horário e se o sensor está localizado em área de rodízio. Caso o veículo que passe no

sistema de identificação, esteja em área e horário não permitidos, de acordo com o final de sua placa, ele pode ser multado automaticamente.

3. DESCRIÇÃO DO FUNCIONAMENTO

Neste item, faremos a descrição do funcionamento do sistema de fiscalização de veículos.

O sistema de fiscalização de veículos proposto possui os seguintes módulos:

- Captura de informação do veículo.
- Transmissão dos dados adquiridos.
- Processamento de dados.
- Exibição do status do veículo.

Na figura abaixo, podemos observar o fluxograma do sistema de identificação de veículos proposto:

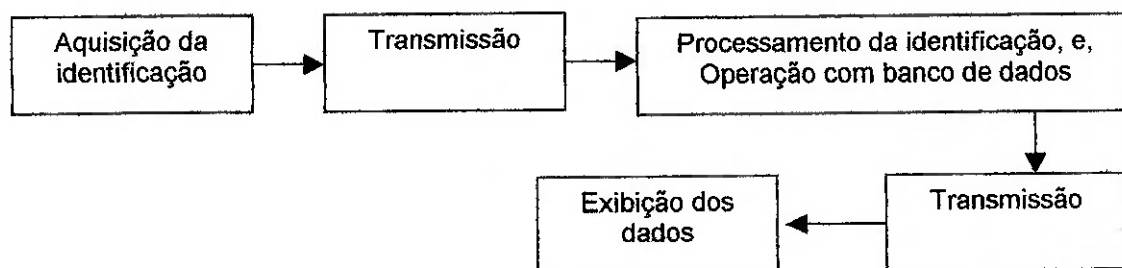


Figura 4: Fluxograma do sistema proposto.

Inicialmente, o veículo em trânsito passa por um ponto de controle, onde através de sensores, é colhido algum tipo de identificação do veículo. A identificação do veículo pode ser feita através da placa de licença do veículo, de um código em um transponder, entre outros. Esta identificação é registrada, e enviada a uma central de dados. Esta central de dados recebe a identificação do veículo, processa esta informação se necessário, e realiza uma busca em seu banco de dados com a finalidade de verificar se existe alguma irregularidade associada a este veículo.

Caso exista alguma irregularidade associada a este veículo, é registrada esta informação, e a mesma pode ser enviada a um ponto de controle mais a frente na via de circulação de veículos ou proximidades da mesma, com a finalidade de que este veículo seja interceptado pela fiscalização policial e as devidas providências sejam tomadas.

Na figura a seguir, podemos ver a ilustração do sistema de fiscalização proposto:

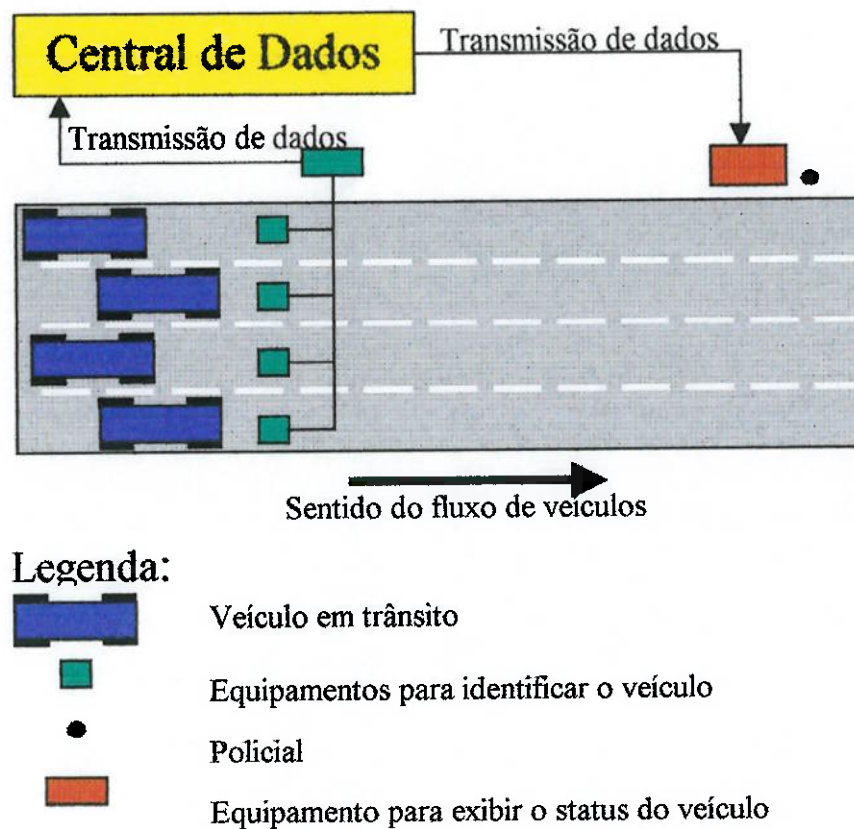


Figura 5: Sistema de fiscalização proposto.

4. DESCRIÇÃO DOS MÓDULOS DO SISTEMA DE FISCALIZAÇÃO DE VEÍCULOS

Neste tópico, estarão descritos os módulos do sistema de veículos, com suas funções e características necessárias.

4.1. Módulo de Captura de Identificação do Veículo

Este módulo tem a função de realizar a captura da informação necessária para realizar a detecção do veículo.

Neste projeto, inicialmente estão sendo estudadas duas formas para a captura da informação necessária para a identificação do veículo, que são:

- Identificação por processamento de imagem.
- Identificação utilizando transponder.

4.1.1. Identificação por Transponder

Um dos modos de realizar a identificação dos veículos é empregar a tecnologia dos transponders. O funcionamento dele é análogo aos sensores de aproximação (Smart Card) utilizados para controle de acesso em prédios sofisticados.

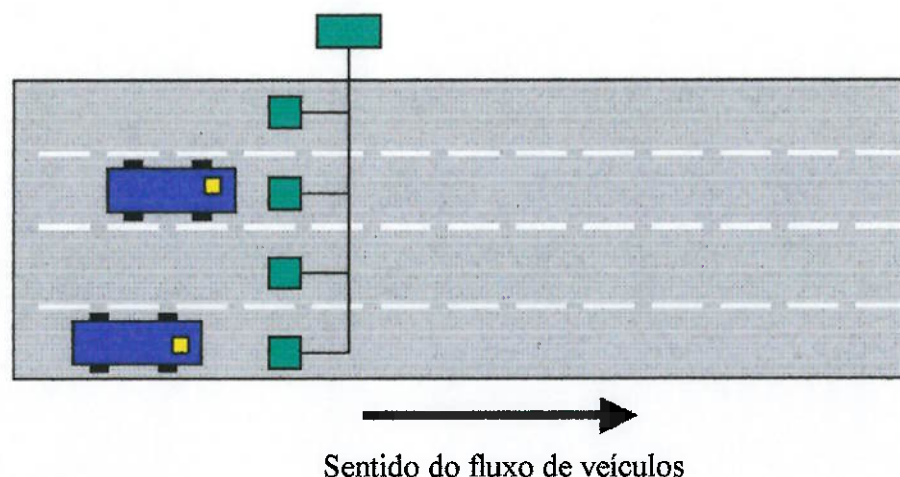
Uma característica que torna o transponder adequado para esta aplicação, é o fato dele ser um componente passivo, ou seja, não necessita de alimentação elétrica, dispensando o uso de baterias ou qualquer outra fonte de alimentação.

Alguns modelos possuem uma memória interna, do tipo EPROM, onde podem ser gravados dados.

O dispositivo de leitura do transponder consiste de um transmissor e uma antena receptora. O transmissor emite ondas magnéticas em frequências elevadas (em torno de 2,45 GHz). Esta frequência não interfere no funcionamento de outros aparelhos eletro-eletrônicos, pois estes últimos operam em frequências muito mais baixas, em torno de 0,4-0,5 GHz.

O transponder é excitado pelo sinal emitido pelo transmissor, e emite o conteúdo de sua memória. A antena do dispositivo de leitura capta esta informação, e transmite esta informação para um sistema de computadores, geralmente computadores pessoais (PC), via porta serial (RS 232, RS 485).

A figura a seguir ilustra o funcionamento de um sistema de transponder:



Legenda:

- Veículo em trânsito
- Dispositivo de leitura do transponder
- Transponder

Figura 6: Sistema de transponder.

A tecnologia utilizando transponders pode ser aplicada a qualquer tipo de veículo. Cada veículo possui um ou mais transponders, que podem ser fixados em diversos pontos do veículo, como, por exemplo: motor, câmbio, teto, chassis, etc.

Em caso de utilização de mais de um transponder por veículo, estes devem estar gravados com a mesma informação, de forma que sejam identificadores únicos do veículo. A utilização de mais de um transponder por veículo tem como vantagem menor possibilidade de erro na identificação do veículo e maior dificuldade em adulterar o veículo. Como desvantagem, aumenta o custo de implantação do sistema, devido o aumento do número de transponders.

Todos os veículos devem possuir um código de identificação, de tal modo que seja um identificador único como, por exemplo: a placa do veículo, o número do chassis, ou um identificador a ser especificado.

Cada ponto de identificação possui um dispositivo de leitura de transponder para realizar a identificação do veículo.

O veículo equipado com transponder ao passar por um ponto de fiscalização será identificado e, esta informação irá para uma central de dados, que verificará se o veículo está em situação regular ou irregular.

4.1.2. Identificação por Processamento de Imagem

Sistemas de reconhecimento automático da placa do veículo possuem muitas aplicações em controle e monitoramento do tráfego de veículos, por exemplo: encontrar veículos roubados, controlar o acesso a estacionamentos, adquirir informações sobre o fluxo de carros, etc.

As técnicas de visão por computador são especialmente úteis em casos em que não é viável obrigar os motoristas a instalar *transponders* nos seus automóveis, pois o reconhecimento do veículo é feito através da própria placa do mesmo.

O sistema utiliza uma câmera para retirar uma imagem da frente ou de trás do veículo. Após isto, um software processador de imagens analisa a mesma e extrai a informação sobre a placa do veículo.

Na Figura 6 mostra-se um esquema simplificado da configuração mínima habitualmente utilizada no sistema de leitura de placas de matrícula, na Universidade de Bristol. O sistema é modular, possuindo todos os módulos um *transputer*, que funciona simultaneamente como elemento de processamento e como controlador do módulo. Os módulos comunicam entre si através dos *links* dos *transputers* e através de um *bus* digital para transferência de imagens (o *bus* pode transferir imagens completas ou somente regiões de interesse). Na configuração da Figura 7 utilizam-se três módulos: um módulo de captura e apresentação de imagem (conversão A/D e D/A), um módulo de armazenamento e processamento (*transputer framestore*), e um módulo de inicialização do sistema e de entrada e saída.

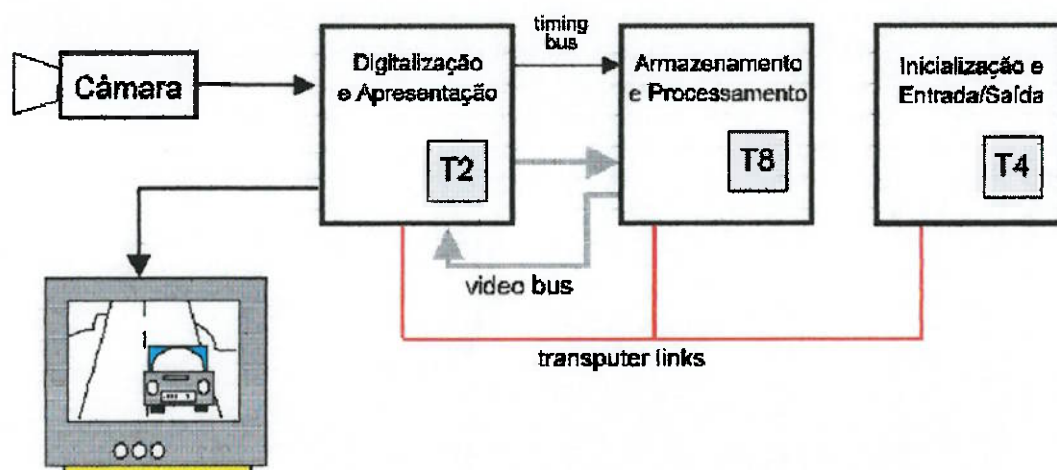


Figura 7: Configuração típica do sistema de leitura de placas de veículos

(Adaptado de License Plate Recognition).

Tendo-se identificado o número da placa do veículo, compara-se esta placa com um banco de dados a fim de saber-se se o veículo possui ou não alguma irregularidade.

Esta técnica possui a vantagem de poder-se registrar uma imagem do veículo e do seu respectivo motorista para eventual identificação posterior, já que a foto do veículo em movimento pode ser armazenada em um computador. Testes feitos com o software de reconhecimento de imagens mostraram que o índice de reconhecimento de placas atinge 98,5% de acerto e leva-se em média 20 ms para a identificação de cada placa.

As técnicas de visão por computador são, no entanto, mais sensíveis às más condições atmosféricas, como nevoeiro e chuva intensa, que podem interferir no reconhecimento da placa.

4.2. Sistema de Transmissão

O sistema de transmissão tem como finalidade transmitir as informações dos pontos de controle, onde a aquisição da identificação do veículo é feita, para a central de dados, onde a informação vai ser processada.

Ela pode ser implementada através de sistemas:

- Via rádio.
- Via par metálico
- Via fibra óptica

Um sistema de transmissão de dados que pode ser utilizado é a implementação de uma rede via Ethernet, utilizando os sistemas de redes privadas, que as operadoras de telecomunicações oferecem.

O dimensionamento da rede depende do número de elementos de rede a serem conectados e da capacidade de transmissão desejada. Cada sensor de identificação é denominado elemento de rede. A capacidade de transmissão geralmente é especificada em bits por segundo (bps), e depende do tipo de dado a ser transmitido e do número de veículos a serem detectados por segundo. Como exemplo, a transmissão de imagens (solução utilizando processamento de imagem) geralmente requer uma maior capacidade de transmissão do que uma sequência de bits gravada em uma memória (solução utilizando transponder).

4.3. Central de Dados

A central de dados é responsável pela recepção da informação sobre a identificação do veículo, o processamento desta informação, busca em banco de dados com a finalidade de verificar se existe alguma irregularidade ou ocorrência associada ao veículo, e caso haja, deverá transmitir os dados do veículo irregular para que sejam tomadas as medidas necessárias.

Caso o sistema possua as funcionalidades de multar automaticamente e/ou rastreamento de veículo, esta central terá a tarefa de processar estes dados de maneira adequada. Estes procedimentos não serão descritos aqui, por serem uma parte opcional do projeto.

O sistema de fiscalização de veículos deverá conter o cadastro de todos os veículos em sua base de dados, contendo suas principais características, como por exemplo: placa, número do chassi, marca, modelo, cor, status, etc. O banco de dados deve estar estruturado da seguinte maneira:

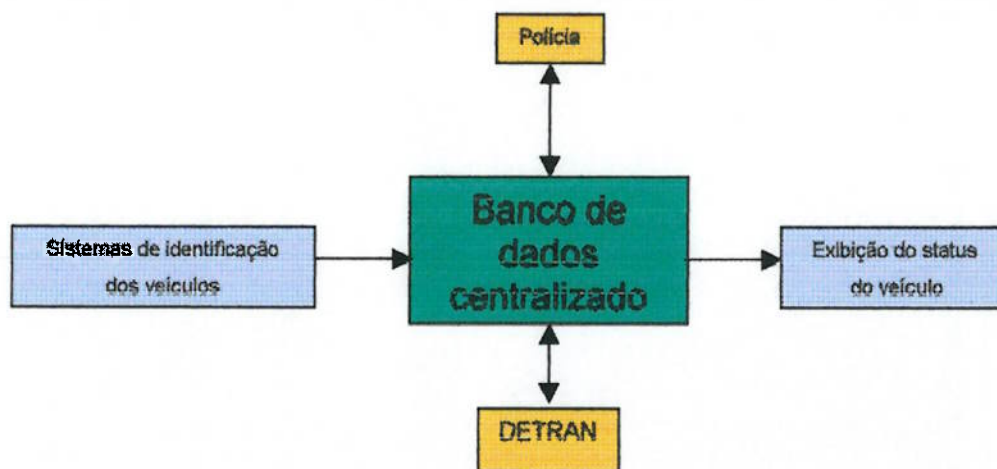


Figura 8: Estruturação do sistema de banco de dados.

O sistema de identificação envia as informações sobre os veículos, para que o sistema realize uma consulta ao banco de dados com a finalidade de verificar se existe alguma irregularidade associada a este veículo. Após este procedimento, o sistema retorna o status do veículo, alertando para alguma irregularidade.

A polícia deve ter acesso ao banco de dados, para que ela registre as ocorrências policiais associadas a um veículo, como por exemplo: roubo, furto, seqüestro, etc.

O DETRAN deve ter acesso ao banco de dados para registrar os veículos novos e realizar consultas e modificações sobre o status de um determinado veículo.

4.4. Módulo de Exibição de Status de um Veículo

Este módulo tem como função fornecer detalhes sobre o veículo irregular, de modo que o policial fiscal possa identificar o mesmo no meio do tráfego.

As informações que este sistema disponibilizará são:

- Tipo de irregularidade associada ao veículo;
- Placa do veículo;
- Modelo do veículo;

- Marca do veículo;
- Cor do veículo.

Estas informações deverão poder ser disponibilizadas através de um computador portátil.

5. SELEÇÃO DO SISTEMA DE AQUISIÇÃO DA IDENTIFICAÇÃO

Este item tem como objetivo a escolha do sistema de aquisição de identificação do veículo. Inicialmente, foram propostos dois sistemas distintos de aquisição da identificação do veículo, já descritos anteriormente:

- Identificação por transponder.
- Identificação por processamento de imagem.

A escolha de um sistema de aquisição da identificação do veículo adequado é fundamental para o sucesso do projeto do sistema de fiscalização, pois os projetos do módulo de transmissão de dados e de processamento de dados estarão diretamente ligados a escolha do sistema de identificação do veículo.

5.1. O Método Escolhido

O método escolhido para realizar a escolha do sistema de aquisição da identificação foi a Matriz de Decisão. A Matriz de Decisão direciona o engenheiro a analisar mais profundamente cada uma das alternativas (soluções), em relação aos critérios do projeto, sem exigir investimentos de tempo e dinheiro com modelamentos matemáticos, desenhos de conjunto e/ou fabricação e protótipos, muitas vezes construídos de maneira precipitada.

A Matriz de Decisão é composta de uma coluna com critérios de seleção, uma com pesos e outra com as notas atribuídas a cada uma das opções disponíveis de projeto, em suas respectivas colunas. Finaliza-se a escolha optando-se pelo projeto que atingir a maior nota. Convém ressaltar que quanto maior for o detalhamento dessa matriz, com relação aos parâmetros apresentados, melhor será a escolha final. Além disso, a determinação dos pesos é fundamental, pois são eles quem caracterizam a importância de cada item relacionado pelo projetista.

Convencionou-se que as notas estariam restritas ao intervalo [1:5], sendo que a atribuição da nota 5 (cinco) corresponde a máxima satisfação do parâmetro (atributo). As notas serão aplicadas comparativamente, ou seja, serão atribuídas em função da comparação do desempenho dos sistemas propostos.

Os pesos estariam restritos aos valores [1,2,3], de acordo com a importância do critério de seleção, sendo que a atribuição do peso 3 (três) corresponde a um critério muito relevante para o sistema de aquisição.

5.2. Critérios de Seleção

Neste item, serão apresentados e descritos os critérios utilizados para a realização da matriz de decisão. Os critérios de decisão foram estabelecidos de forma a avaliar os processos de operação, implantação e manutenção do sistema. Foi levada em conta a segurança proporcionada pelo sistema contra tentativas de falsificações.

Capacidade de reconhecimento do sistema - Este critério avalia a capacidade dos sistemas propostos de realizar a aquisição do identificador em condições normais. O peso atribuído a este critério é 3 (três).

Nível de segurança contra falsificações - Este critério avalia o nível de segurança proporcionado pelo sistema de identificação contra tentativas de burlar o sistema de fiscalização de veículos. O peso atribuído a este critério é 3 (três).

Interferência de fenômenos da natureza - Este critério avalia a interferência de fenômenos da natureza no processo de aquisição da identificação. Os fenômenos da natureza levados em conta neste critério são chuva, neblina e noite (falta de luminosidade). O peso atribuído a este critério é 3 (três).

Facilidade de instalação do identificador - Este critério avalia a facilidade em realizar a instalação do identificador na frota de veículos a serem monitorados. Serão considerados os veículos novos e a frota de veículos usados. O peso atribuído a este critério é 2 (dois).

Manutenção do sistema - Este critério avalia o período em que o sistema opera sem a ocorrência de fatos que alterem a eficiência do sistema. O peso atribuído a este critério é 2 (dois).

Tráfego de dados na rede de comunicação gerado por cada identificação - Este critério avalia a quantidade de volume de tráfego ocasionado pelo sistema de identificação. O módulo de transmissão de dados será projetado em função deste tráfego gerado. O peso atribuído a este critério é 1 (um).

Necessidade de processamento dos dados adquiridos - Este critério avalia a necessidade de processamento dos dados adquiridos. O módulo de processamento de dados será projetado em função desta necessidade. O peso atribuído a este critério é 1 (um).

Capacidade do sistema de aquisição simultânea de identificadores - Este critério avalia a capacidade do sistema de aquisição simultânea de vários identificadores. Este é necessário em via de múltiplas faixas de trânsito, como por exemplo, a pista da Marginal Pinheiros, em frente à Cidade Universitária, onde existem sete faixas trânsito. O exemplo anterior é um caso extremo. Normalmente os principais corredores de trânsito na cidade de São Paulo possuem em média quatro faixas de trânsito. O peso atribuído a este critério é 1 (um).

A seguir será apresentada uma tabela com um resumo dos critérios de seleção estabelecidos, com seus respectivos pesos:

Tabela 1: Critérios de Seleção.

Critério de Seleção	Peso Atribuído
Capacidade de reconhecimento do sistema	3
Nível de segurança contra falsificações	3
Interferência de fenômenos da natureza	3
Facilidade de instalação do identificador	2
Manutenção do sistema	2
Tráfego de dados na rede gerado por cada identificação	1
Necessidade de processamento dos dados adquiridos	1
Capacidade do sistema de aquisição simultânea de identificadores	1

Soma Total dos Pesos:	16
------------------------------	-----------

5.3. Definição da Matriz de Decisão

Neste item será definida a escolha da melhor solução para o sistema de aquisição da identificação, através do método da Matriz de Decisão. A seguir será realizada a definição das notas do sistema em função dos critérios estabelecidos anteriormente.

Para a alternativa de reconhecimento por transponder (alternativa 1), foi usado como referência o transponder HIPERXTM, modelo LVM 4033, do fabricante Balogh.

Para a alternativa de reconhecimento por processamento de imagem (alternativa 2), foi usado como referência o modelo estudado na Universidade de Bristol.

Capacidade de reconhecimento do sistema

Alternativa 1 - Identificação por transponder: Nesta alternativa, o sistema possui 99,99% de sucesso na identificação, ou seja, ele não consegue identificar em média 1 transponder a cada 10.000 requisição de reconhecimento. Esta taxa é obtida para a velocidade de referência de 100 km/h.

Nota atribuída a este critério: 5 (cinco).

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, o sistema possui 98,5% de sucesso na identificação, sendo que este resultado não leva em consideração a velocidade do veículo, pois foi obtido com o veículo parado.

Nota atribuída a este critério: 2 (dois).

Nível de segurança contra falsificações

Alternativa 1 - Identificação por transponder: Nesta alternativa, é muito difícil ocorrer tentativas de burlar o sistema, já que a identificação é feita através de uma leitura por radiação eletromagnética. Para ocorrer a falsificação, seria necessária a

troca do chip, pois a memória do transponder é uma EPROM, ou seja, uma vez gravada, não pode ser apagada ou regravada. Como o transponder deve ser trocado a cada 10 anos, no momento da troca, pode haver falsificação do chip, caso o processo não seja rigorosamente controlado.

Nota atribuída a este critério: 4 (quatro).

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, o método de adulterações já é conhecido pelos criminosos, sendo que estas alterações são facilmente produzidas até mesmo por um cidadão comum (tapamento da placa, forjamento de letras e números falsos com fita isolante, raspamento dos caracteres da placa com a utilização de ferramentas, etc).

Nota atribuída a este critério: 1 (um).

Interferência de fenômenos da natureza

Alternativa 1 - Identificação por transponder: Nesta alternativa, a leitura do transponder é feita através de radiação eletromagnética. Neste processo, a eficiência da aquisição da identificação (leitura do transponder) não é alterada devido aos fenômenos da natureza, como por exemplo, chuva, neve, neblina ou noite (falta de luminosidade).

Nota atribuída a este critério: 5 (cinco).

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, fenômenos meteorológicos podem interferir na capacidade de identificação da placa do veículo em questão, basta que estes fenômenos afetem as condições de visibilidade da via em que o sistema de reconhecimento esta sendo empregado.

Nota atribuída a este critério: 2 (dois).

Facilidade de instalação do identificador

Alternativa 1 - Identificação por transponder: Nesta alternativa, a identificação do veículo seria feita através da leitura de alguns dados armazenados em um transponder. O transponder estaria fixado no motor e no câmbio dos veículos. Isto é

justificado pelo fato de serem peças caras para serem trocadas. Nos veículos novos, o transponder seria instalado pelas montadoras. No caso da frota de veículos usados, seria necessário realizar um processo para a instalação dos mesmos, o que ocasionaria alguns transtornos para donos de veículos e autoridades.

Nota atribuída a este critério: 2 (dois)

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, a identificação do veículo é feita através da placa de licença do veículo. O sistema de identificação de veículos através de sua placa de licença não necessitaria de alterações. Com isto, não haveria nenhuma modificação na frota de veículos usados e os procedimentos em relação aos veículos novos não precisariam ser alterados.

Nota atribuída a este critério: 5 (cinco)

Manutenção do sistema

Alternativa 1 - Identificação por transponder: O transponder possui uma vida média de 10 anos, ou seja, após este período, o transponder deverá ser substituído. O processo de substituição do transponder não é trivial, e pode aumentar as chances de ocorrer fraudes no sistema. O nível de sinal emitido pelo transponder ao longo de sua vida útil é constante, ou seja, não há variação do índice de identificação do veículo enquanto seu transponder estiver em operação.

Nota atribuída a este critério: 2 (dois).

Alternativa 2 – Identificação por processamento de imagem: As placas de trânsito sofrem desgaste natural dependendo da exposição do veículo em que estas estão empregadas ao tempo e ao ambiente (radiação solar, chuvas, salinidade do ar de cidades costeiras, etc). O tempo de vida de uma placa de trânsito, esta, portanto relacionado com o local de uso do veículo em que esta está empregada. No entanto, mudanças de proprietários dos veículos podem ocasionar mudanças nas placas dos mesmos, dependendo da vistoria realizada no veículo durante o ato da mudança de proprietário. Portanto, em média, podemos estimar a vida útil da placa do veículo tão longa quanto a própria vida útil do veículo.

Nota atribuída a este critério: 4 (quatro).

Tráfego de dados na rede de comunicação gerado por cada identificação

Alternativa 1 - Identificação por transponder: Nesta alternativa, cada antena de identificação gera um tráfego na rede de 30 kbit/s. Para uma rede de telecomunicações, esta taxa de transmissão é muito baixa.

Nota atribuída a este critério: 5 (cinco).

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, o tráfego de rede é praticamente composto por imagens de vídeo que possuem o tamanho aproximado de 600 kbits cada, sendo que a câmera capta 16 quadros por segundo. Portanto, a taxa de transmissão de vídeo para uma câmera é equivalente à 16 vezes 600 kbits/s, o que equivale à 9600 kbits/s.

Nota atribuída a este critério: 1 (um).

Necessidade de processamento dos dados adquiridos

Alternativa 1 - Identificação por transponder: Nesta alternativa, a identificação do veículo é feita através dos dados armazenados no transponder. Não há a necessidade de processar estes dados para que seja feita a identificação do veículo.

Nota atribuída a este critério: 5 (cinco).

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, após o recolhimento da imagem da placa de um determinado veículo, ainda é necessário que se utilize um software de OCR para que extraiam-se os caracteres presentes na placa de matrícula para que estes possam ser utilizados para o reconhecimento do veículo em questão.

Nota atribuída a este critério: 1 (um).

Capacidade do sistema de aquisição simultânea de identificadores

Alternativa 1 - Identificação por transponder: Nesta alternativa, o sistema de leitura dos transponders pode operar com até 31 canais distintos. Com isto, pode ser realizada a leitura de até 31 transponders simultaneamente, sendo uma leitura para

cada antena. Além disso, cada antena pode realizar a leitura de até 5 transponders no intervalo de 1 segundo.

Nota atribuída a este critério: 5 (cinco).

Alternativa 2 – Identificação por processamento de imagem: Nesta alternativa, o sistema de leitura é dependente do número de câmeras disponíveis para a fiscalização das vias de tráfego, sendo que cada câmera deve ser responsável por apenas uma faixa de tráfego de veículos.

Nota atribuída a este critério: 3 (três).

5.4. Matriz de Decisão

A matriz de decisão pode ser vista na tabela abaixo:

Tabela 2: Matriz de Decisão.

		SOLUÇÕES PROPOSTAS			
		SOLUÇÃO COM TRANSPONDER		SOLUÇÃO COM PROCESSAMENTO DE IMAGEM	
ATRIBUTOS	Peso	Nota	NxP	Nota	NxP
Capacidade de reconhecimento do sistema	3	5	15	2	6
Nível de segurança contra falsificações	3	4	12	1	3
Interferência de fenômenos da natureza	3	5	15	2	6
Facilidade de instalação do identificador	2	2	4	5	10
Manutenção do sistema	2	2	4	4	8
Tráfego de dados na rede gerado por cada identificação	1	5	5	1	1
Necessidade de processamento dos dados adquiridos	1	5	5	1	1
Capacidade do sistema de aquisição simultânea de identificadores	1	5	5	3	3
SOMA (S)	16	—	65	—	38
NOTA FINAL (S/16)		—	4,06	—	2,38

6. REQUISITOS DO SISTEMA

Neste item, serão definidos os requisitos operacionais do sistema, ou seja, estará sendo especificado como o sistema de fiscalização automática de veículos deverá funcionar. O sistema implantado deverá atender a todos os requisitos estabelecidos a seguir:

- Área de aplicação do sistema: Para a implantação inicial do sistema, fica estabelecida a área metropolitana da cidade de São Paulo. Este dado será considerado para avaliar as condições climáticas e infra-estrutura existentes.
- Método para a identificação de veículos: De acordo com a matriz de decisão realizada previamente, a tecnologia escolhido para a identificação automática de veículos é a solução por transponder.
- Capacidade de transmissão: O sistema de transmissão deverá comportar todo o tráfego de dados gerado pelo sistema de aquisição da identificação dos veículos e pela central de dados. Os valores estarão sendo determinados no item 8.
- Tempo máximo para a transmissão: O sistema de identificação deverá realizar todas as transmissões de dados e seu respectivo processamento em um tempo não superior a 7,2 segundos.
- Confiabilidade do sistema: O sistema de identificação automática de veículos deverá ter uma confiabilidade de no mínimo 95%, ou seja, é aceitável uma falha a cada 20 identificações.
- Velocidade máxima de identificação do veículo: Considerando-se a aplicação inicial, fica estabelecida que o sistema de fiscalização automática deverá identificar veículos até a velocidade de 100 km/h. Este valor foi dimensionado de acordo com as velocidades máximas permitidas na região metropolitana de São Paulo (90km/h).
- Índice de irregularidade dos veículos: O índice atual de veículos com irregularidade (pendência judicial, documentação vencida) na cidade de São Paulo é de 25% (Fonte: Jornal da Tarde – 28 de Julho de 2002).
- Frota de veículos na cidade de São Paulo: A frota atual de veículos na cidade de São Paulo é de 5.200.000 (2002e).

7. PROJETO DO MÓDULO DE CAPTURA DE IDENTIFICAÇÃO DO VEÍCULO

7.1. Introdução

O sistema de fiscalização automática de veículos escolhido consiste em um sistema de identificação sem contato, utilizando a tecnologia de identificação por tag.

O projeto do sistema de fiscalização automática de veículos está baseado no sistema de identificação Hyper.X, modelo LM_4033 fabricado pela Balogh Group.

A Balogh Group é uma empresa francesa especializada no desenvolvimento e fabricação de sistemas de identificação. Entre as aplicações para os seus sistemas de identificação desenvolvidos estão o controle de acesso, controle de produção, controle de estoque, entre outras aplicações que necessitem de um grande número de identificação de pessoas ou objetos.

O sistema de identificação Hyper.X funciona em altas frequências (2,45 GHz), e foi originalmente desenvolvido pela Thomson CSF Communications, e foi adquirido pela Balogh Group no ano de 1999.

O sistema de identificação automática de veículos é composto por:

- Transponder.
- Sistema de leitura.
- Sistema de processamento local.
- Sistema de processamento central.
- Sistema para a exibição dos dados.

7.2. Ambiente de Operação

Neste tópico, serão analisadas as compatibilidades entre as condições ambientes especificadas para o funcionamento do sistema de identificação e as condições ambientes encontradas na cidade de São Paulo, onde seria instalado o sistema piloto de identificação automática de veículos.

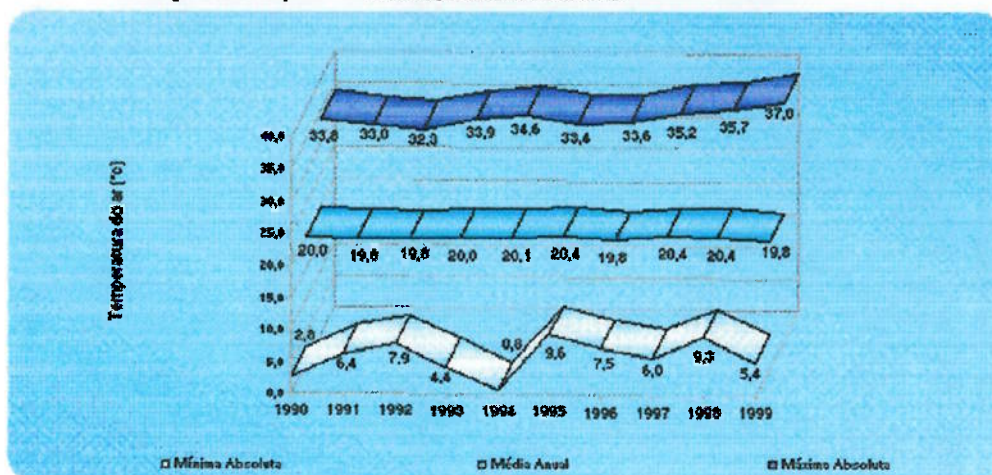
7.2.1. Faixa de Temperatura de Operação

O sistema de identificação automática de veículos deve estar instalado em um ambiente com as seguintes condições de temperatura:

- Temperatura mínima de operação: -20°C
- Temperatura máxima de operação: $+70^{\circ}\text{C}$

A evolução da temperatura na cidade de São Paulo pode ser vistas no gráfico abaixo:

Gráfico 1.1 Município de São Paulo
Evolução das Temperaturas Médias, Máximas e Mínimas Anuais: 1990/1999



Fonte: Instituto Nacional de Meteorologia - INMET (Vil Distrito), 1990/1999.

Elaboração: Emplasa, 2000.

Figura 9: Evolução das temperaturas médias e mínimas anuais da cidade de São Paulo.

No gráfico acima são fornecidas as temperaturas máxima, mínima e média ocorridas na cidade de São Paulo. Deste gráfico, encontramos os valores de temperaturas máximo e mínimo ocorrido na década de 90 na cidade de São Paulo.

- Temperatura mínima ocorrida na cidade de São Paulo: $+0,8^{\circ}\text{C}$ (1994)
- Temperatura máxima ocorrida na cidade de São Paulo: $+37^{\circ}\text{C}$ (1999)

Podemos concluir que o sistema escolhido é adequado à temperatura de São Paulo.

O sistema possui proteção contra poeira e água nível IP 55. Este nível garante proteção contra poeira, mas a poeira pode penetra, mas não em quantidade suficiente para prejudicar a operação do sistema. Este nível garante proteção a jatos de água em qualquer direção. Para maiores detalhes, consulte o apêndice 1.

7.2.2. Umidade Relativa

O sistema de identificação de veículos não deve operar por longos períodos em ambientes onde a umidade relativa seja superior a 90%.

7.3. Tranponder

O transponder, também conhecido como tag, é um dispositivo cuja função é armazenar uma informação, pela qual, será possível a identificação do veículo. Portanto, a informação armazenada deve ser um identificador único. Cada veículo possuirá um transponder fixado a ele.

O transponder utilizado no sistema de fiscalização automática de veículos é o modelo BDG_1090 fabricado pela Balogh Group. As suas principais características são:

- Dimensões: 70 x 50 x 20 mm
- Peso: 80 g
- Temperatura: -40 / +85° C
- Grau de proteção: IP67
- Proteção contra erro: HDLC
- Taxa de erro: 10^{-7}
- Capacidade total: 180 bits/ 30 caracteres.
- Capacidade disponível: 162 bits/ 27 caracteres.
- Frequência: 2,45 GHz
- Taxa de transmissão: 30 Kbit/s
- Velocidade máxima de operação: 100 Km/h
- Tempo de vida: + 10 anos

As ondas eletromagnéticas na frequência de 2,45 GHz permite uma alta capacidade de transmissão de dados. A detecção do tag é realizada de uma maneira muito rápida e é pouco susceptível as interferências do ambiente de operação. Fora do alcance da leitora, o transponder é eletromagneticamente inativo. Sua única função é a capacidade de refletir as microondas incidentes. O transponder refletirá a portadora incidente em 2,45 GHz, modulada pela sua identificação individual, em direção a leitora. É possível a identificação simultânea de vários transponders em uma mesma área de detecção.

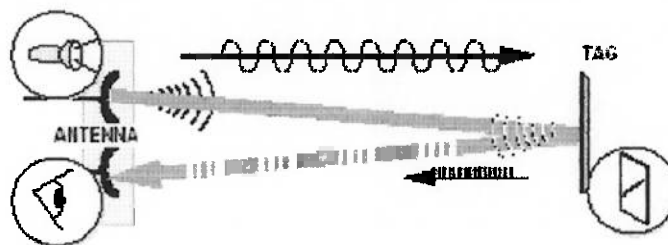


Figura 10: Princípio de funcionamento do tag (Adaptado de Descritivo Técnico Balogh).

7.3.1. Codificação do Transponder

O transponder possui capacidade para armazenar até 180 bits, ou seja, até 30 caracteres. O sistema de codificação utiliza 3 caracteres para uso interno, por isso, temos a capacidade disponível de armazenamento de até 27 caracteres.

Os formatos de dados suportados para o armazenamento são:

- ASCII
- ISO2
- WIEGAND

O padrão de caracteres utilizados na codificação do transponder será o ASCII, padrão que suporta caracteres alfanuméricos. O identificador único de cada veículo será o seu número de chassi. Será através deste que o veículo será identificado.

A identidade de um veículo é a numeração do chassi, que possui 17 dígitos. Os três primeiros números são a identificação internacional do fabricante. O primeiro se refere à área geográfica em que o veículo foi produzido (8 e 9 representam a América do Sul). Já o segundo dígito especifica o País em que o carro foi fabricado (B quer dizer Brasil e A, Argentina, por exemplo). O terceiro identifica o fabricante (F, Ford; W, Volkswagen; D, Fiat; G, GM, etc.). Entre o 4.º e o 9.º dígitos está o quadro descritivo do carro, como tipo de veículo, motor e potência, equipamentos de segurança e o nome do modelo, entre outros. O 10.º dígito representa o ano de fabricação do veículo e o 11.º, o local da fábrica. O restante dos números significa a individualidade, ou seja, o número de série do carro.

Até 1.982, não existia na marcação do chassi uma descrição detalhada do veículo, expressa na forma de letras e números. Cada montadora gravava os seus produtos com a quantidade de caracteres que melhor lhe aprouvesse, sem que houvesse uma amarração com o veículo fisicamente.

Em 1.985, o CONTRAN (Conselho Nacional de Trânsito), através da Resolução 659/85 (publicada no DOU em 30/10/85), regulamentou a marcação do chassi, criando algumas normas, entre elas a de 17 caracteres.

Desta forma, definimos o seguinte padrão para a codificação do transponder:

- 1ª a 3ª posição: Reservado para uso interno.
- 4ª a 20ª posição: Reservado para o número do chassi do veículo.
- 21ª a 27ª posição: Não utilizado.

7.4. Sistema de Leitura

O sistema de leitura é realizado através do sistema de identificação Hyperx, modelo LVM_4033, fabricado pela Balogh Group e é composto dos seguintes módulos:

- Emissor de sinais microondas.
- Antena receptora.
- Fonte de Alimentação.
- Sistema de processamento local

As principais características operacionais do sistema de fiscalização automática escolhida são:

- | | |
|---|------------------|
| • Frequência de operação: | 2,45 GHz |
| • Número de canais: | 31 |
| • Taxa de transmissão entre o leitor e o tag: | 30k bit/s |
| • Modo de detenção de erro: | HDLC |
| • Taxa de erro nas identificações: | 10^{-7} |
| • Potência irradiada: | 75 mW E.I.R.P. |
| • Temperatura de operação: | - 20° C à +70° C |
| • Temperatura de armazenamento: | - 25° C à +80° C |
| • Nível de proteção contra poeira e água: | IP 55 |
| • Alimentação: | 10 à 15 VDC |
| • Velocidade máxima de operação: | 100 km/h |

7.4.1. Sistema de Alimentação

O módulo do sistema de aquisição de informação é alimentado por uma fonte de tensão contínua (10 à 15 VDC). O sistema proposto trabalhará com sistema de alimentação redundante. Desta maneira, em caso de falha no fornecimento de energia da fonte principal, o sistema de segurança automaticamente acionará a fonte de energia reserva, não prejudicando o funcionamento do sistema.

A fonte de alimentação principal será fornecida pela rede de distribuição de energia elétrica convencional, mediante a utilização de um transformador e a fonte de alimentação secundária será fornecida por baterias auxiliares.

O esquema de ligação pode ser visualizado na figura abaixo:

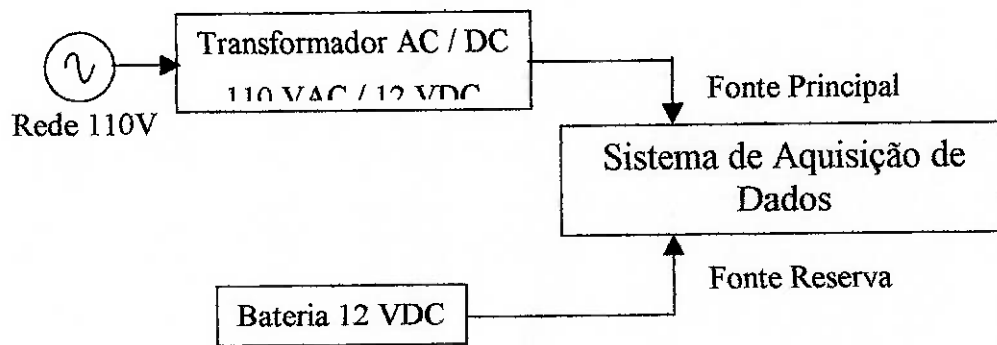


Figura 11: Projeto da alimentação do sistema de aquisição de dados.

7.4.2. Emissor de sinais microondas e antena receptora.

O sistema emissor de sinais microondas e antena receptora é parte do sistema de aquisição de dados. Estes dois componentes trabalham em conjunto, de modo a realizar a aquisição dos dados armazenados no transponder. O sistema emissor de sinais de microondas possui a finalidade de transmitir uma portadora em uma frequência específica (2,45 GHz), de modo a excitar o transponder. A antena receptora tem a função de captar o sinal refletido pelo transponder e possibilitar a recepção dos dados de identificação do veículo.

A figura abaixo ilustra este processo:

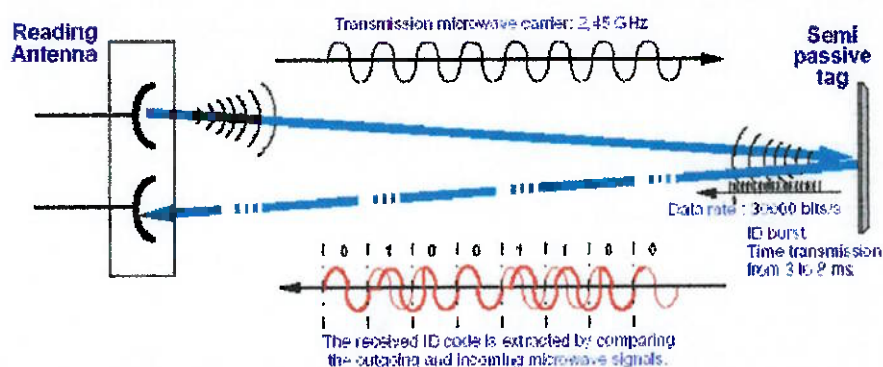


Figura 12: Funcionamento da antena (Adaptado de Descritivo Técnico Balogh).

A partir da comparação do sinal recebido e da portadora emitida, é possível o reconhecimento dos dados armazenados no transponder. Esta identificação é negativa, ou seja, a identificação dos bits é realizado de maneira inversa. Caso o bit armazenado no tag seja 1, será detectado o bit 0 e vice versa.

Com relação ao sinal detectado pela antena receptora, caso o bit armazenado em uma determinada posição do tag seja 1, não haverá defasagem no sinal. Caso o bit armazenado no tag seja 0, haverá um atraso no sinal. Através desta comparação, é possível a aquisição dos bits armazenados na transponder.

A figura abaixo ilustra o processo de identificação de um veículo:

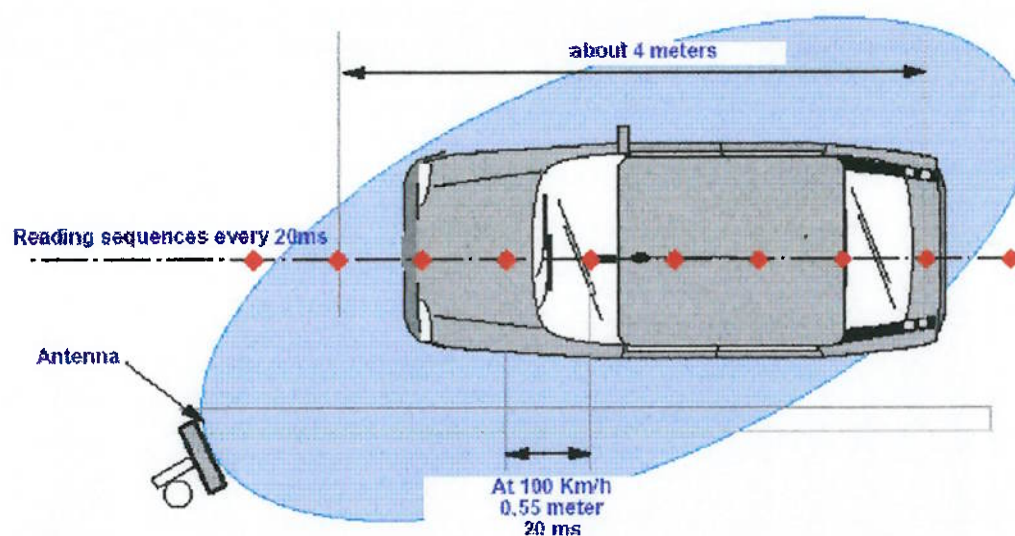


Figura 13: Captação da identificação do veículo pela antena (Adaptado de Descritivo Técnico Balogh).

O alcance da antena pode variar em função da directividade da antena. Com isto, no momento da instalação deve ser escolhido a posição e a directividade mais adequada em função das condições da instalação. A próxima figura ilustra este fato:

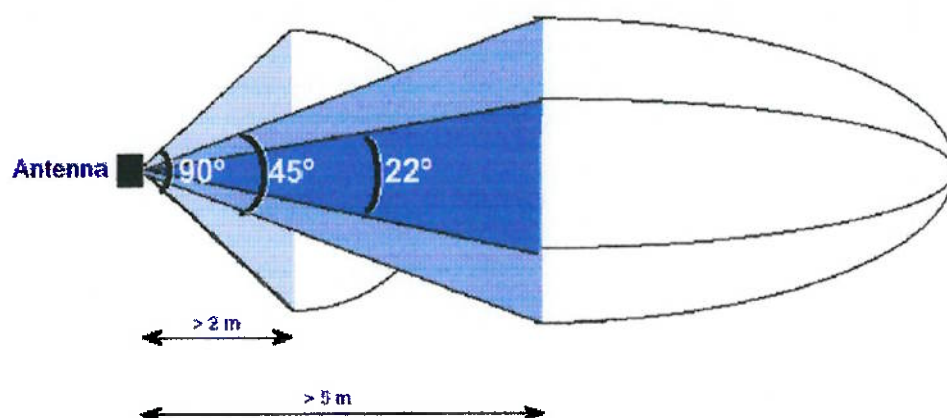


Figura 14: Directividade da antena (Adaptado de Descritivo Técnico Balogh).

Com uma directividade de 90°, podemos alcançar uma distância máxima de 2 metros. Esta situação é adequada quando é necessário monitorar apenas uma faixa de

trânsito. Com uma directividade de 45° a 22° , a antena possui um alcance de 5 metros.

A seguir serão ilustrados algumas possibilidades para o posicionamento da antena:

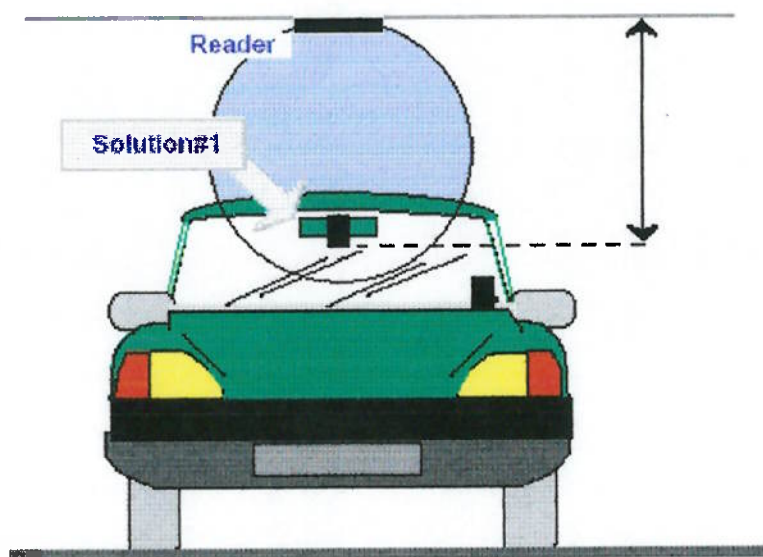


Figura 15: Antena posicionada acima dos veículos I (Adaptado de Descritivo Técnico Balogh).

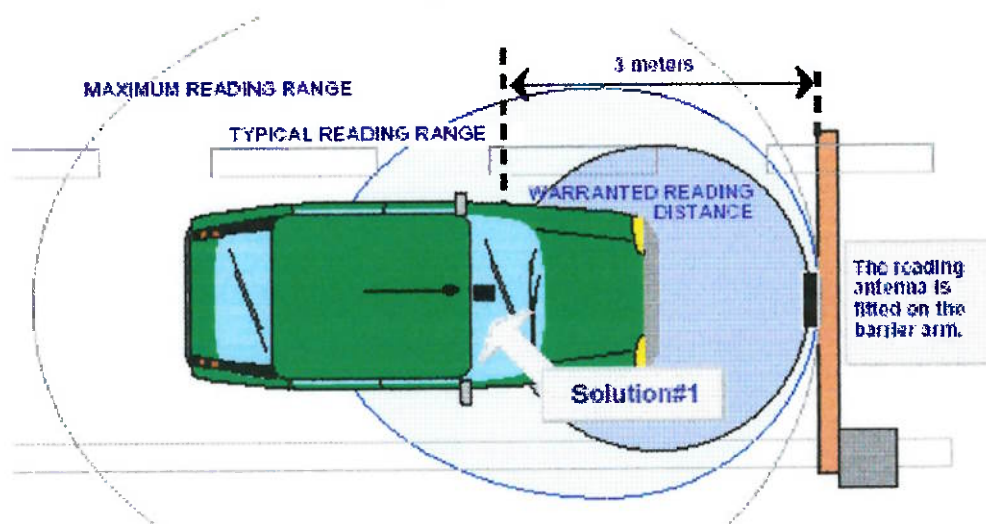


Figura 16: Antena posicionada acima dos veículos II (Adaptado de Descritivo Técnico Balogh).

Nos exemplos acima, a identificação é realizada através do sistema de recepção localizado em uma braço, que passa por cima da via. Esta solução é adequada utilizando-se directividade de 90° e pode ser aplicada em vias de faixa de trânsito simples ou múltiplas.

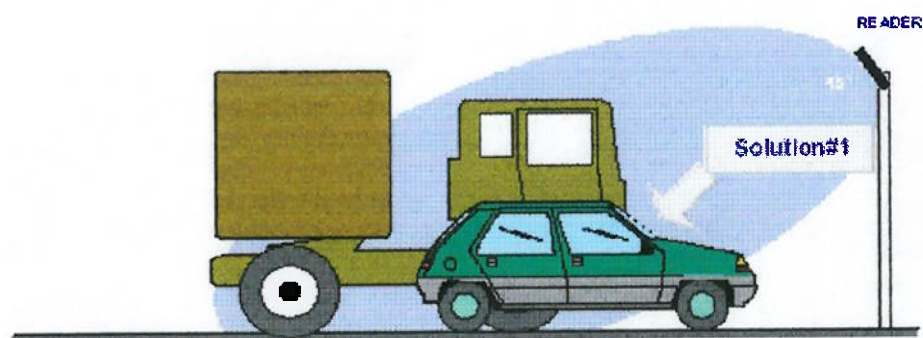


Figura 17: Antena posicionada lateralmente I (Adaptado de Descritivo Técnico Balogh).

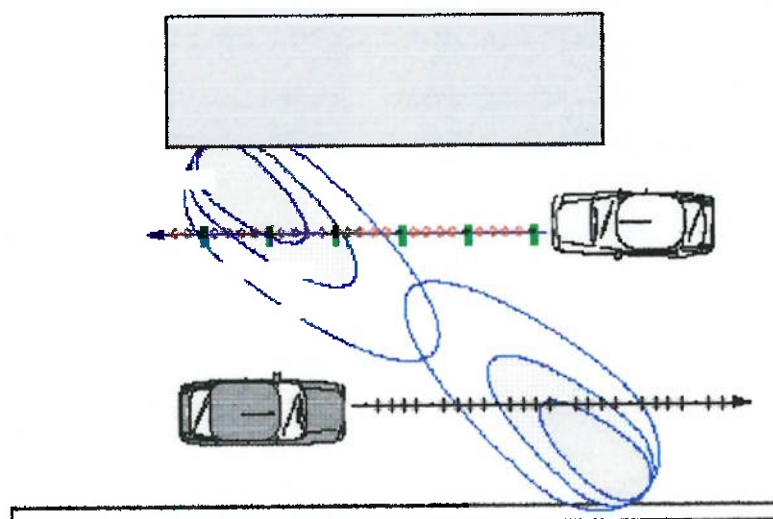


Figura 18: Antena posicionada lateralmente II (Adaptado de Descritivo Técnico Balogh).

Nos dois exemplos acima, a antena é posicionada lateralmente, com posicionamento semelhante aos aparelhos de fiscalização de velocidade fixos em

operação na cidade de São Paulo. A aplicação desta configuração é restrita para vias com poucas faixas de trânsito e a directividade adequada é de 22° .

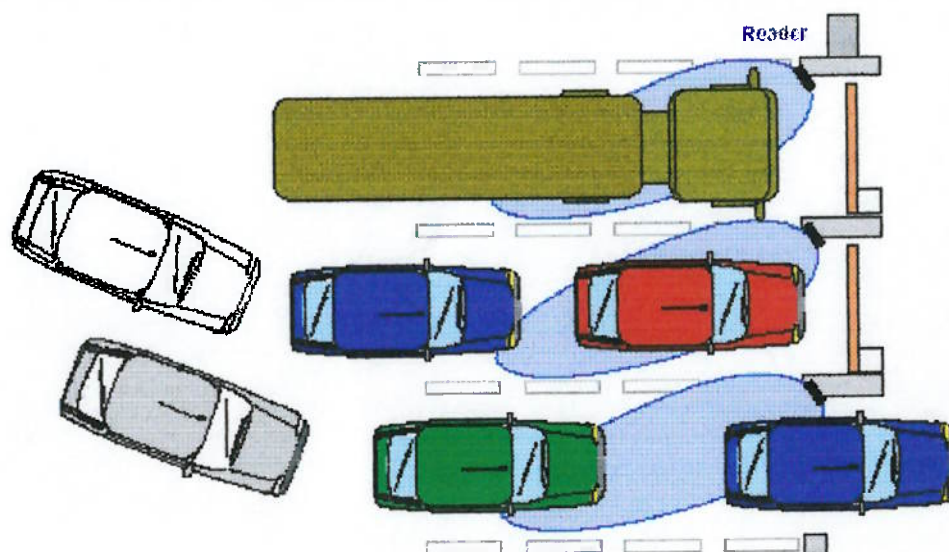


Figura 19: Antena posicionada lateralmente para múltipla faixas (Adaptado de Descritivo Técnico Balogh).

Na configuração apresentada na figura anterior, cada sistema de identificação é posicionado lateralmente a cada faixa de trânsito. Com isto é possível a aplicação desta configuração em vias de múltiplas faixas de rolamento. A directividade adequada para este tipo de configuração é de 45° . A sua desvantagem é a necessidade de uma separação física entre as faixas de rolamento.

7.4.3. Sistema de Processamento Local.

Após a captação dos dados realizado pela antena, é necessário o pré processamento destas informações. Cada pré processador pode controlar até duas antenas. A arquitetura deste pré processador central pode ser visto abaixo:

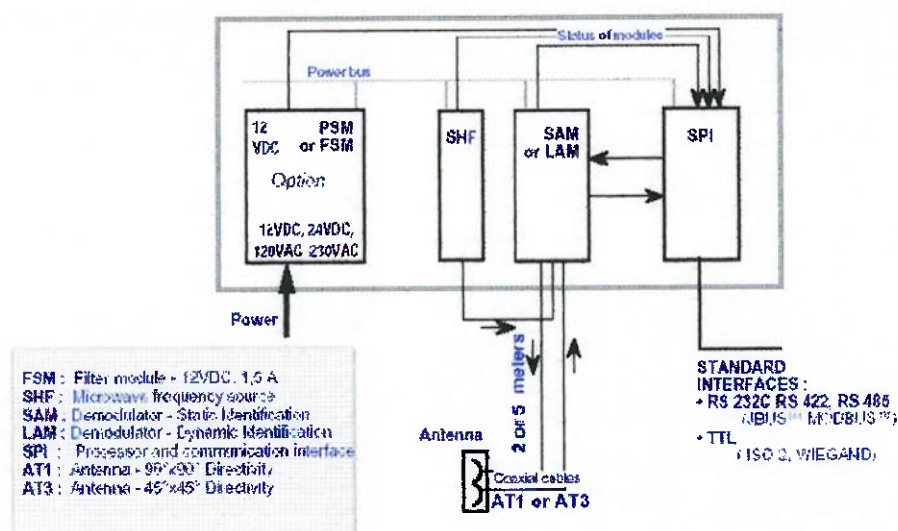


Figura 20: Arquitetura do sistema de aquisição de dados (Adaptado de Descritivo Técnico Balogh).

Pela figura acima podemos observar que o sistema de pré processamento possui quatro módulos, que são:

- Filtro de linha (FSM): Módulo responsável pela entrada de energia elétrica e distribuição deste para os outros módulos.
- Gerador de microondas (SHF): Módulo responsável pela geração de microondas. Este sinal é enviado para o módulo LAM.
- Demodulador (LAM): Módulo responsável pela interface com o transmissor de microondas e com a antena.
- Processador e interface de comunicação: Módulo responsável pelo processamento das informações dos três módulos anteriores. Executa também a interface com o processador central das informações.

Após o pré processamento das informações, estes são enviados para o sistema de processamento central.

7.4.4. Interconexão do Sistema de Processamento Local e Antena

A interconexão do sistema de processamento central e da antena é feito através de um par de cabos coaxiais. A distância máxima desta conexão pode variar, de acordo com o tipo de cabo utilizado.

A próxima figura ilustra este processo:

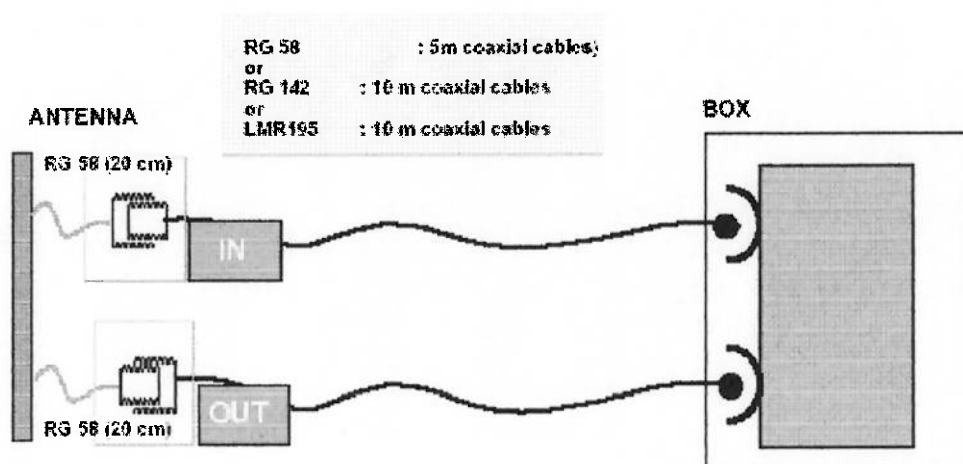


Figura 21: Conexão entre o sistema de processamento local e a antena

(Adaptado de Descritivo Técnico Balogh).

Tabela 3: Tipo de cabo X Máxima distância de transmissão

Tipo de cabo	Distância máxima de transmissão (m)
RG 58	5
RG 142	10
LMR 195	10

7.5. Sistema de Processamento Central

O sistema de processamento central é responsável pelo processamento das informações dos sistemas pré processamento (Local Unit) sob seu controle. Os requisitos mínimos para este sistema são:

- Pentium 200 MHz.
- Memória 8 Mb RAM.
- Windows 95.

Cada sistema de processamento central pode controlar até 12 unidades locais. A comunicação entre as unidades locais e a unidade central é feita através de um barramento serial.

A figura abaixo ilustra o sistema de identificação de veículos:

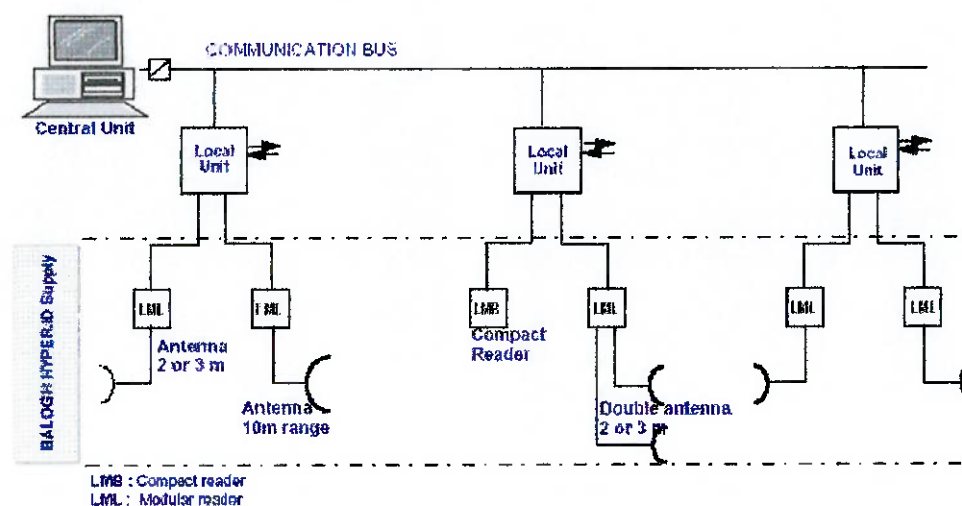


Figura 22: Sistema de processamento de dados central (Adaptado de Descritivo Técnico Balogh).

A comunicação serial pode ser realizada de diferentes formas:

- **RS-232:** Este protocolo deve ser utilizado apenas para comunicação ponto-a-ponto. No caso da unidade central controlar mais que uma unidade local, este protocolo não pode ser utilizado.
- **RS-422:** Este protocolo pode ser utilizado para comunicações ponto-a-ponto e ponto-multiponto. Este protocolo utiliza dois pares de cabos (Rx e Tx).
- **RS-485:** Este protocolo pode ser utilizado para comunicações ponto-a-ponto e ponto-multiponto. Este protocolo utiliza apenas um par de cabo (Rx e Tx).

As ilustrações destes tipos de comunicação podem ser vistos abaixo:

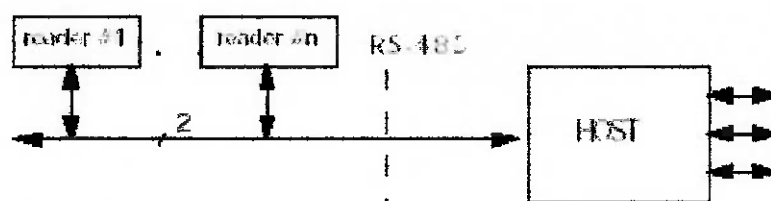


Figura 23: Configuração ponto-multiponto - RS 485.

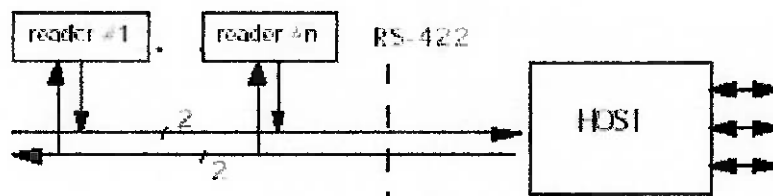


Figura 24: Configuração ponto-multiponto - RS 422 (Adaptado de Descritivo Técnico Balogh).

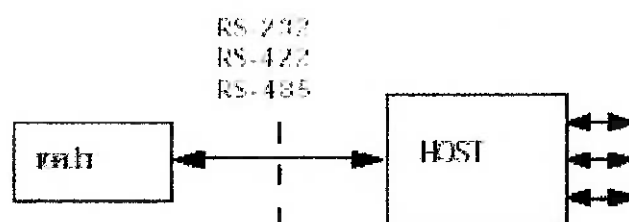


Figura 25: Configuração ponto-a-ponto – RS232, RS 422, RS-485 (Adaptado de Descritivo Técnico Balogh).

A central de processamento de dados envia o número do chassi do veículo para a central de dados, que receberá este dado e acessará o banco de dados. Esta transmissão será feita através de placas Ethernet. O dimensionamento deste sistema de transmissão está realizado mais à frente.

Os seguintes protocolos podem ser utilizados:

- 7 bits / paridade par / 1 stop bit.
- 7 bits / paridade ímpar / 1 stop bit.

- 8 bits / sem paridade / 1 stop bit.

A porta serial do computador pode ser configurada em quatro velocidades:

- 1200 bps
- 2400 bps
- 9600 bps
- 19200 bps

8. DIMENSIONAMENTO DO SISTEMA DE TRANSMISSÃO

8.1. Premissas

O sistema de transmissão de dados é responsável pelo transporte das informações coletadas no sistema de aquisição de dados para o sistema de processamento central de dados. O dimensionamento será realizado considerando-se o pior caso de operação.

O sistema será dimensionado levando-se em conta os seguintes critérios:

- Capacidade de transmissão
- Tempo para a transmissão
- Confiabilidade
- Facilidade de implantação

Os dados utilizados para realizar o dimensionamento do sistema de transmissão podem ser visto abaixo:

- Velocidade máxima para identificação do veículo: 100 km/h.
- Considerando o sistema de aquisição de dados operando em sua capacidade máxima.
- Cada antena é responsável pela detecção de 2 faixas de trânsito.
- Distância máxima entre o ponto de aquisição e o ponto de controle: 300m.
- Fator de segurança: 2.
- Confiabilidade: 99%
- Veículos em situação irregular: 25%.

Com os dados acima, temos que o tempo máximo para se realizar o processamento e a transmissão dos dados é de 7,2 segundos.

O sistema de transmissão está dividido em duas partes distintas:

- Parte I: Transmissão dos dados do sistema de aquisição de dados para a central de dados.
- Parte II: Transmissão dos dados dos veículos em situação irregular da central de dados para o ponto de fiscalização.

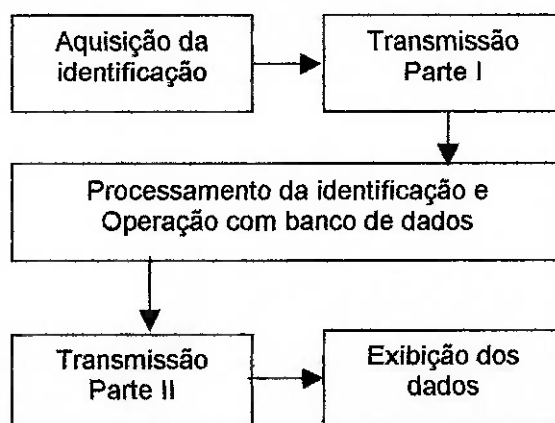


Figura 26: Fluxograma do sistema de fiscalização automática de veículos.

8.2. Sistema de transmissão: Sistema de aquisição de dados - Central de dados

Neste item, será projetado o sistema de transmissão de dados do sistema de aquisição de dados dos veículos para a central de dados. Para o dimensionamento deste sistema de transmissão, será considerada a utilização do equipamento em sua capacidade máxima e um fator de segurança 2.

O sistema de leitura em sua capacidade máxima, possui a seguinte configuração:

- 12 módulos de leitura.
- 2 antenas por módulo de leitura.
- 6 identificações/segundo por antena (velocidade de 100km/h / Detecção de duas faixas de trânsito para cada antena).

O sistema de leitura, operando dentro dos pré-requisitos em sua capacidade máxima é capaz de realizar até 144 identificações por segundo.

$$Identificações/Segundo = 12 \text{ módulos de leitura} \times 2 \frac{\text{antenas}}{\text{módulo de leitura}} \times 6 \frac{\text{Identificações}}{\text{Antena} \cdot \text{Segundo}} = 144$$

Temos que cada identificação (transmissão do número do chassi) requer aproximadamente 50 bytes. A taxa de transmissão requerida, pode ser calculada por:

$$Tx = I \times b \times Fs$$

Sendo: Tx: Taxa de transmissão requerida.

b: Banda requerida por identificação.

Fs: Fator de segurança.

A taxa de transmissão requerida para esta parte da transmissão é de 14,4 Kb/s

8.3. Sistema de Transmissão: Central de dados – Ponto de fiscalização.

Neste item, será projetado o sistema de transmissão de dados da central de dados para o ponto de fiscalização. Neste sistema, serão transmitidos os dados dos veículos irregulares. Para o dimensionamento deste sistema de transmissão, será considerada a utilização do equipamento em sua capacidade máxima, fator de segurança 2 e um índice de veículos irregulares de 25%.

Do item anterior, temos que o número máximo de identificações por segundo do sistema é de 144. Portanto, teremos 36 veículos em situação irregular. Para cada veículo em situação irregular, a central de dados tem que transmitir os seguintes dados:

- Tipo de irregularidade associada ao veículo;
- Placa do veículo;
- Modelo do veículo;
- Marca do veículo;
- Cor do veículo.

Para a transmissão de cada registro, é necessário aproximadamente 70 bytes. A taxa de transmissão requerida, pode ser calculada por:

$$Tx = l \times b \times Fs$$

Onde: Tx: Taxa de transmissão requerida.
b: Banda requerida por identificação.
Fs: Fator de segurança.

A taxa de transmissão requerida para esta parte da transmissão é de 5,04 Kb/s

9. SISTEMA DE TRANSMISSÃO

9.1. Introdução

O sistema de transmissão utilizado será composto por um sistema de rádio ponto-multiponto para coletar os dados do sistema de identificação. Então, este sistema de rádio agrupará estas informações em uma linha privada SDH de 256 kb/s. Foi considerado a mesma infra-estrutura para o sistema de transmissão do sistema de aquisição de dados para a central de dados e da central de dados para o ponto de fiscalização.

Abaixo, segue um diagrama esquemático:

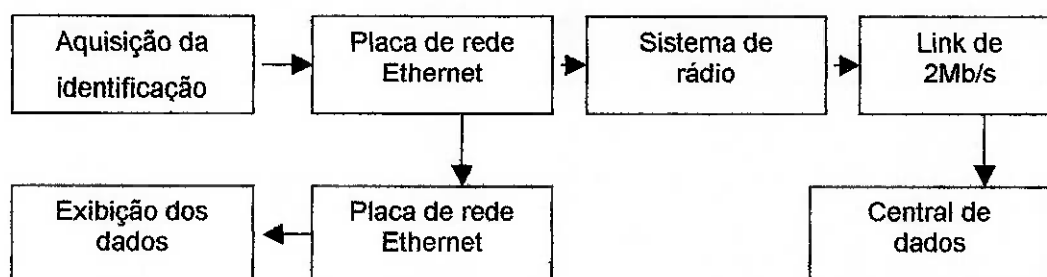


Figura 27: Diagrama em blocos do sistema de transmissão.

9.2. O Sistema de Rádio

O sistema de rádio foi escolhido devido a maior mobilidade que este proporciona. O rádio coleta as informações provindas do sistema de coleta de informações e transmite para um concentrador de tráfego, que utiliza a infra-estrutura das operadoras de telefonia para transmitir os dados para a central de dados.

O sistema de rádio ponto multiponto escolhido foi o BreezeACCESS, fabricado pela Siemens.

O BreezeACCESS é um sistema de acesso sem fio voltado a aplicações de voz e dados a clientes residenciais (condomínios), SOHO (*Small Office Home Office*) e pequenas e médias empresas. Através de uma tecnologia de transmissão de pacotes por meio de acesso sem fio, o sistema BreezeACCESS suporta serviços IP de alta velocidade, incluindo acesso de banda larga à Internet e VPNs (Virtual Private Networks). Os usuários da rede formada

pelo sistema BreezeACCESS podem usufruir de um acesso “always on” suportando taxas de até 3Mbps, além de serviço telefônico.

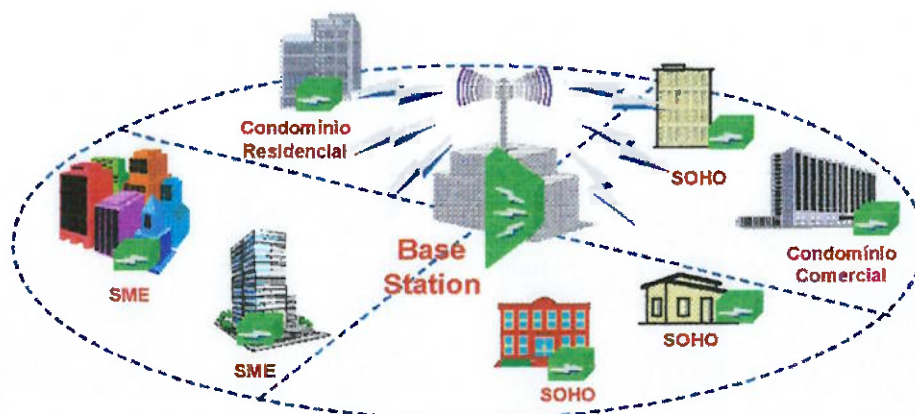


Figura 28: Topologia típica de uma rede banda larga sem fio (Adaptado de Apresentação Técnica Rádio Ponto-Multiponto Siemens).

O sistema é chamado ponto multiponto, devido uma base station concentrar o tráfego de vários subscriber unit.

O subscriber unit fica localizado junto ao sistema de aquisição de dados, e transmite os dados via rádio para a base station. O sistema de aquisição de dados transmite os dados através de uma interface Ethernet padrão 10BaseT. O padrão 10BaseT é suficiente para atender a demanda de tráfego exigida pelo sistema de aquisição de dados. A figura abaixo mostra a conexão entre o processador de dados do sistema de aquisição de dados e o subscriber unit do sistema de transmissão de rádio:

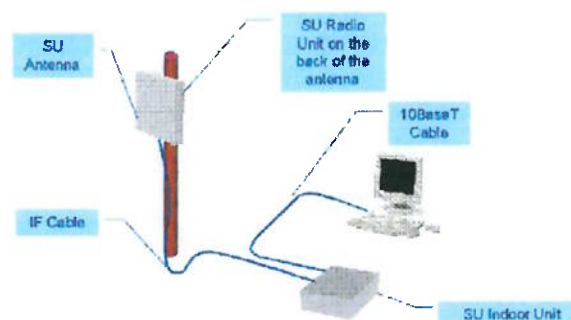


Figura 29: Ligação entre o sistema de aquisição de dados e o subscriber unit (Adaptado de Apresentação Técnica Rádio Ponto-Multiponto Siemens).

A base station pode receber e/ou enviar o dados para até seis subscriber units. Com isto, o tráfego máximo efetivo do sistema é de 86,4 kbit/s. A base station se conecta a rede da operadora de telecomunicações através de um link dedicado de 256 kbit/s simétrico, ou seja, a mesma taxa para download e upload.

O sistema de rádio empregado pode ser visto na figura a seguir:

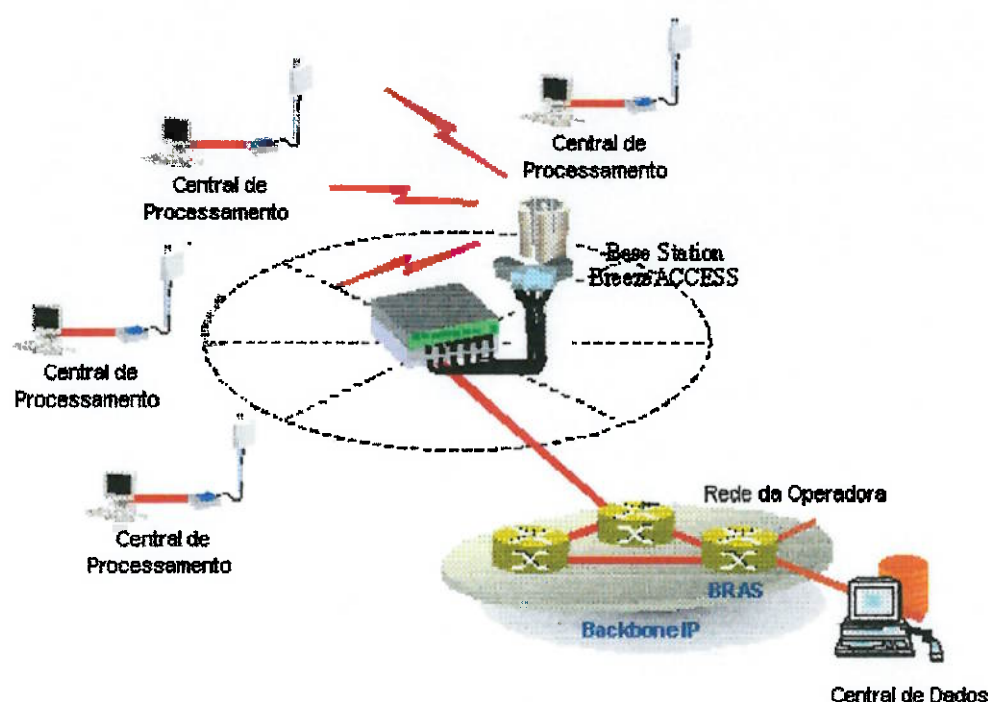


Figura 30: Sistema de transmissão de dados via rádio.

9.2.1. Especificações Técnicas – Sistema de Rádio

9.2.1.1. Especificações Gerais:

Faixa de Frequência:	2400 a 2483,5 MHz
Método de acesso:	FH-CDMA (Frequency Hopping Code Division Multiple Access)
Modulação	GFSK (Gaussian Frequency Shift Keying)
Largura do Canal	1MHz
Potência da terminal	26dBm
Potência da estação base (BS-AU)	26dBm
Sensibilidade (10E-6)	-73dBm (para 3Mbits/s)

9.2.1.2.Base Station

Arranjo Celular	6 setores, com possibilidade de sobreposição de até 3 camadas, resultando em 18 setores
Capacidade	18Mbps (gross rate) para célula de 6 setores 54Mbps (gross rate) para célula com 3 camadas de 6 setores
Número Máximo de Terminais	O número máximo de terminais dependerá basicamente da capacidade total demandada. Um setor pode suportar até 512 estações terminais, resultando num total de 3072 SUs por Base Station de 6 setores
Interfaces	Uma interface 10/100BaseT para cada setor
Alimentação	-48VDC

9.2.1.3.Subscriber Unit

Capacidade Máxima	3Mbps (Gross Rate)
Alimentação	100VAC-240VAC (47-63Hz)
Interfaces	Voz: POTS - conector RJ 11 Dados: Ethernet – conector RJ45
Tipos de terminais	SU-A
Padrões de voz	<u>Voz sobre IP:</u> Padrão ITU-T H.323 v2 <u>Codificação de voz:</u> ITU-T G.711 (64kbps) ITU-T G.723.1 (6,3kbps) ITU-T G.729A (8kbps) <u>Características adicionais:</u> Compressão de Silêncio VAD (Voice Activity detection)
Características adicionais	▪ CIR (Committed Information Rate) e MIR (Maximum Information Rate) definidos para o downlink e o uplink, para cada terminal ▪ Encriptação RC4 ▪ Filtragem de Broadcasting ▪ Filtragem IP de endereços de assinantes e endereços de gerência

9.3. Linha Privada – 256 Kb/s

Após a Base Station do sistema de rádio coletar as informações provindas dos Subscribers Unit, estas informações são transmitidas para a central de dados através de uma linha privada provisionada por uma operadora de telefonia. A taxa de transmissão escolhida foi de 256 kb/s. Esta taxa deve ser simétrica, ou seja, a mesma taxa para download e upload. Para a determinação desta taxa, foi considerado um fator de segurança igual a 3.

Em São Paulo, existem várias operadoras que oferecem este tipo de serviço, como por exemplo:

- AT&T
- DirectNet
- Diveo
- Embratel
- Intelig
- Pégasus Telecom
- Telefônica

A linha privativa contratada deve ter uma taxa efetiva de 256 kb/s, ou seja, deve ser uma linha exclusiva, não compartilhada com outras aplicações.

10. PLANO DE PROJETO DE SOFTWARE

10.1. Descrição Rápida do Funcionamento do Software

Os dados referentes a situação dos veículos circulantes em uma via e dos pontos de controle onde os sensores serão instalados são disponibilizados em um banco de dados.

O Sistema de Fiscalização de Veículos adquire o Código de Acesso Único (Número do Chassi) do veículo circulante em uma via através de sensores.

Com este dado, o sistema faz uma consulta em um banco de dados e verifica se o veículo que circula pela via é irregular ou não.

Caso o veículo que esteja circulando nesta via seja irregular, o mostra as informações referentes a este veículo em um display localizado em um posto de controle da polícia localizado nesta via.

10.2. Requisitos de Projeto de Software

10.2.1. Introdução

Projeto de componentes de software para desenvolvimento do Sistema de Fiscalização de Veículos.

10.2.2. Marcas

Várias marcas e nomes de produtos podem ser citados neste documento e que pertencem a terceiros. Dado que este documento pretende ter circulação apenas interna, não será feita uma discriminação caso a caso.

10.2.3. Escopo

Este documento formaliza a especificação de requisitos de projeto de software, descrevendo suas características básicas, objetivo, funções principais, cronograma de desenvolvimento e implementação.

10.2.4. Conceitos, terminologia e abreviaturas

Tabela 4: Termologia Utilizada no Plano de Projeto de Software

Termo	Significado
S.F.V.	Sistema de Fiscalização de Veículos
HTML	Hyper Text Markup Language
SQL	Structured Query Language
Jboss	Servidor <i>web</i> distribuído gratuitamente pela Sun.

10.2.5. Descrição Geral**10.2.5.1. Perspectiva Histórica**

Ver introdução.

10.2.5.2. Objetivos

O objetivo deste projeto é viabilizar a automação de um protótipo do sistema de identificação de veículos irregulares que circulam em uma via. Porém, para efeito de apresentação o mesmo é dividido em partes:

10.2.5.3. Projeto do Banco de Dados

Durante a execução do sistema, todas as informações necessárias estarão armazenadas em um banco de dados MS Access que deverá ser a fonte de informações utilizada pelo servidor *web* para os processamentos necessários em cada solicitação do cliente.

O projeto do banco de dados envolve muito da inteligência do sistema e desta forma existe um item totalmente dedicado à explanação das tabelas utilizadas e do modelo relacional construído.

10.2.5.4. Desenvolvimento dos Aplicativos

O sistema será construído em linguagem Java, pois visa-se a operação do mesmo via rede, como a internet por exemplo, e desta forma optou-se por esta linguagem que é amplamente utilizada atualmente e que também é uma plataforma bastante robusta.

10.2.5.5. Design das Interfaces do Sistema com o Usuário

Existirão telas de interface para o usuário poder entrar com todos os dados necessários ao funcionamento do programa e também telas de saída das informações dos veículos irregulares que eventualmente passaram por um ponto de controle do S.F.V.

Estas serão escritas em HTML devido a este sistema poder ser operado via rede, como dito anteriormente. Note que as mesmas permitem a correta interação do cliente com o aplicativo, pois encaminham os *inputs* a serem fornecidos.

10.2.5.6. Requisitos Funcionais

Cadastro

Os dados referentes aos veículos irregulares circulantes serão armazenados em uma tabela. Em uma outra tabela, serão armazenados os dados referentes aos locais onde serão colocados os pontos de fiscalização automática dos veículos.

O sistema deverá: permitir ao operador cadastrar novos veículos irregulares e novos locais de fiscalização, eliminar aqueles veículos que já não são mais irregulares ou que tenham sido retirados de circulação e os locais de fiscalização que não estejam mais sendo utilizados, e, listar os veículos irregulares e os locais de fiscalização cadastrados.

O sistema poderá, conforme a necessidade: permitir a visualização de veículos irregulares e pontos de fiscalização cadastrados a partir de um código ou de uma sequência de caracteres que identifique as placas dos veículos irregulares ou endereço dos pontos de localização cadastrados ou de outro atributo que facilite a operação do usuário.

Identificação de Veículos Irregulares Circulantes

O S.F.V. colhe através de sensores os dados referentes aos veículos que transitam pelas vias onde estão localizados os pontos de fiscalização automática.

O sistema deverá: adquirir, através do sinal dos sensores, o código ou outro atributo de identificação dos veículos circulantes na via onde esta localizado o ponto

de fiscalização automática, e, identificar os veículos que são irregulares e que estão trafegando nesta via.

Registro de Passagem de Veículo Irregular

O sistema poderá: registrar em uma tabela: o código ou atributo de identificação do veículo irregular, o local do ponto de fiscalização automática e a data da passagem do veículo, de forma a registrar a ocorrência da passagem de veículo irregular por certa via de circulação.

Disponibilização das Informações de Veículos Irregulares

O S.F.V. disponibilizará em um *display* as informações referentes e relevantes sobre os veículos irregulares que circulam em uma determinada via onde esteja havendo fiscalização automática de forma a facilitar o trabalho de identificação destes veículos por parte das autoridades competentes.

O sistema deverá: disponibilizar em uma tela de computador as informações referentes e relevantes sobre os veículos irregulares que circulam em uma via próxima a este *display*.

Consulta Online

A consulta online permite que usuários ligados em rede obtenham informações referentes aos veículos irregulares.

O sistema poderá: permitir a consulta por parte dos usuários através do código do veículo ou uma sequência de caracteres que identifique as placas de determinados veículos aos veículos que estão cadastrados no sistema.

10.2.5.7. Requisitos Não-Funcionais

Ambiente de Execução

A máquina utilizada será....sdaufa

O sistema operacional será o MS Windows que contem o MS Explorer como *browser* para utilização do servidor instalado e da visualização das páginas HTML.

O próprio processador representa o cliente e o servidor, embora possa ser montado um conjunto de processadores para demonstração da tecnologia *servlet*.

O servidor é o JBOSS, proveniente do site da Sun de forma gratuita.

Interfaces com Outros Sistemas

Este sistema funciona via rede e por isto pode ser acessado de outro PC que esteja conectado ao servidor simbolizando um cliente qualquer.

Outros sistemas poderiam acessar o banco de dados deste através de comandos SQL ou de saídas que atingissem as páginas HTML desenvolvidas neste trabalho. Funcionalmente, porém, não é previsto nenhum tipo de interface com outros sistemas.

Desempenho

O desempenho do sistema vem do poder de processamento da máquina utilizada e esta relacionado também com o tamanho do banco de dados. O sistema deverá responder, caso o veículo identificado seja irregular, de forma que este ainda não tenha passado pela posição em que esta o *display* na via.

Escalabilidade

A escalabilidade do número de pontos de fiscalização de controle depende apenas da instalação física destes pontos de controle, uma vez que o Sistema de Fiscalização de Veículos e o seu banco de dados estão disponibilizados via rede.

Segurança e Privacidade

A segurança do sistema é um dos requisitos fundamentais do mesmo. A linguagem Java permite a criação e utilização de chaves públicas e privadas e o controle de acesso de um indivíduo ou grupo de indivíduos à estas chaves e aos certificados.

O sistema poderá possuir pelo menos dois níveis de segurança:

- Administrador: Capaz de operar todas as funcionalidades do sistema;

- Usuário (polícia rodoviária): Restrito a observação de registros de veículos identificados como irregulares e a consultas de veículos cadastros.

10.3. Modelagem do Banco de Dados e Tabelas Access:

Começaremos definindo a estrutura do banco de dados, ou seja, as tabelas de informações que este deverá conter.

10.3.1. Tabela de Informações sobre o Veículo e sua Documentação (veículos)

Esta tabela será a tabela de referência para o sistema de controle. Ela conterá as informações necessárias para identificar se o veículo está regularizado ou não. Deverá conter:

Tabela 5: veículos: Tabela de Informações sobre o Veículo e sua Documentação

Nome	Nome Abreviado	Tipo de Dados
Número do chassi do veículo (chave da tabela)	NumChassi	CHAR(17)
Código do Renavam do veículo	CodRenavam	CHAR(10)
Nome do proprietário do veículo	NomeProp	CHAR(45)
CPF / CGC do proprietário do veículo	CpfProp	CHAR(11)
Placa do veículo e UF	Placa	CHAR(9)
Marca / Modelo do veículo	Modelo	CHAR(20)
Espécie / Tipo do veículo	TipoVeic	CHAR(20)
Ano de fabricação / Ano do modelo do veículo	AnoFabr	CHAR(9)
Cor predominante do veículo	CorVeic	CHAR(10)
Veículo roubado / furtado	Roubado	CHAR(1)
Documentação Irregular	DocIrreg	CHAR(1)

Número do chassi do veículo: este campo trará o número do chassi do veículo e será o código que identifica unicamente um veículo dentro da tabela.

Código do Renavam do veículo: código que identifica unicamente um veículo no território nacional.

Nome do proprietário do veículo: este campo conterá o nome do proprietário do veículo.

CPF / CGC do proprietário do veículo: este campo conterá o CPF ou CGC do proprietário do veículo.

Placa do veículo e UF: este campo conterá a placa do veículo e a unidade federal de procedência do mesmo.

Marca / Modelo do veículo: este campo conterá a marca e o modelo do veículo.

Espécie / Tipo do veículo: este campo conterá a espécie e o tipo do veículo.

Ano de fabricação / Ano do modelo do veículo: este campo trará o ano de fabricação e o ano do modelo do veículo.

Cor predominante do veículo: este campo conterá a cor predominante do veículo.

Veículo roubado ou furtado: neste campo constará se o veículo está ou não constando como roubado ou furtado nos arquivos da polícia.

Documentação Irregular: neste campo constará se o veículo está ou não irregular com relação à documentação do mesmo.

10.3.2. Tabela de Informações sobre os Pontos de Fiscalização de Veículos (locais)

Esta tabela será a tabela que conterá informações sobre as localidades dos pontos de fiscalização automática de veículos. Deverá conter:

Tabela 6: locais: Tabela de Informações sobre os Pontos de Fiscalização de Veículos

Nome	Nome Abreviado	Tipo de Dados
Código de identificação do ponto de fiscalização de veículos	CodFisc	CHAR(7)

Local do ponto de fiscalização de veículos	EndFisc	CHAR(45)
--	---------	----------

Código de identificação do ponto de fiscalização de veículos: é o código que identifica unicamente um ponto de fiscalização de veículos.

Local do ponto de fiscalização de veículos: é a localização, endereço, onde se encontra um determinado equipamento para a fiscalização automática de veículos.

10.4. Transações:

Conforme escrito anteriormente, o aplicativo utilizará da tecnologia *servlet* para sua execução. Abaixo, segue uma breve descrição do aplicativo e das funções que compõem o mesmo. As listagem das mesmas estão localizadas no Anexo.

O sistema funciona da seguinte forma:

Através da rede, o banco de dados pode ser acessado e modificado para inclusão, alteração e exclusão de registros de veículos e locais de pontos de fiscalização.

Desta forma, tem-se um banco de dados acessível via rede para consulta de registros de veículos.

O sinal de controle chega das antenas para os sensores e é passado para os computadores dos postos de fiscalização. Deste ponto, estes arquivos são abertos e o aplicativo *servlet* faz uma consulta ao banco de dados para saber se algum veículo irregular passou pelas antenas dos sensores naquele momento. Se ocorreu de um veículo irregular ter passado pelas antenas naquele momento, o *servlet* retorna ao computador do posto de fiscalização localizado à frente àquela antena uma janela mostrando os dados principais para a identificação deste veículo irregular por parte do policial de plantão.

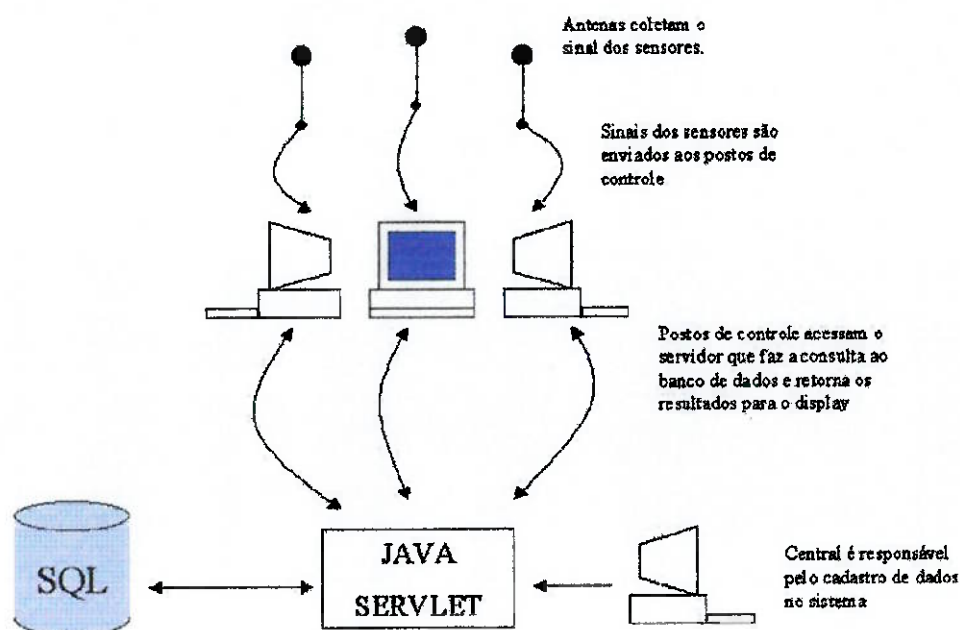


Figura 31: Esquema de Funcionamento do Programa do S.F.V..

InsereVeículo:

Descrição:

Insere um registro na tabela de veículos.

Entradas:

Informações do veículo e do proprietário do veículo.

Saídas:

Inserção de um registro de um veículo dentro da tabela “veiculos”.

Mensagem de operação realizada com sucesso ou de erro, conforme o caso.

Entidades e atributos referenciados:

Veiculos: Todos os atributos

Procedimentos armazenados utilizados:

Outros processamentos:

Consulta Veiculo:*Descrição:*

Consulta um registro na tabela de veículos.

Entradas:

Informações do veículo.

Saídas:

Registro do veículo encontrado ou mensagem de registro não encontrado, quando for o caso.

*Entidades e atributos referenciados:**Procedimentos armazenados utilizados:**Outros processamentos:***Altera Veiculo:***Descrição:*

Altera um registro na tabela de veículos.

Entradas:

Número do Chassi do veículo em questão.

Saídas:

Mensagem de operação realizada com sucesso ou de erro, conforme o caso.

Entidades e atributos referenciados:

Veiculos: Todos os atributos

*Procedimentos armazenados utilizados:**Outros processamentos:***Deleta Veiculo:***Descrição:*

Apaga um registro na tabela de veículos.

Entradas:

Número do Chassi do veículo em questão.

Saídas:

Mensagem de operação realizada com sucesso ou de erro, conforme o caso.

Entidades e atributos referenciados:

Veiculos: Todos os atributos

Procedimentos armazenados utilizados:

Outros processamentos:

InserLocal:

Descrição:

Inserir um registro na tabela de locais.

Entradas:

Informações do local de fiscalização.

Saídas:

Inserção de um registro de um local de fiscalização na tabela "locais".

Mensagem de operação realizada com sucesso ou de erro, conforme o caso.

Entidades e atributos referenciados:

Locais: Todos os atributos

Procedimentos armazenados utilizados:

Outros processamentos:

ConsultaLocal:

Descrição:

Consulta um registro na tabela de locais.

Entradas:

Informações do local de fiscalização.

Saídas:

Registro do local de fiscalização encontrado ou mensagem de registro não encontrado, quando for o caso.

Entidades e atributos referenciados:

Procedimentos armazenados utilizados:

Outros processamentos:

AlterLocal:

Descrição:

Altera um registro na tabela de local.

Entradas:

Código de identificação do local de fiscalização em questão.

Saídas:

Mensagem de operação realizada com sucesso ou de erro, conforme o caso.

Entidades e atributos referenciados:

Locais: Todos os atributos

*Procedimentos armazenados utilizados:**Outros processamentos:***DeletaLocal:***Descrição:*

Apaga um registro na tabela de locais.

Entradas:

Código de identificação do local de fiscalização em questão.

Saídas:

Mensagem de operação realizada com sucesso ou de erro, conforme o caso.

Entidades e atributos referenciados:

Locais: Todos os atributos

*Procedimentos armazenados utilizados:**Outros processamentos:***ProcuraVeiculo:***Descrição:*

Consulta na tabela veículos se o veículo é irregular ou não a partir da entrada do sinal proveniente do sensor.

Entradas:

Número do Chassi do veículo em questão.

Saídas:

Identificação do veículo em questão.

Mostra em uma janela do windows no local de controle os principais dados do veículo em questão caso este veículo seja irregular.

Entidades e atributos referenciados:

Veículos: Todos os atributos

Procedimentos armazenados utilizados:

Outros processamentos:

10.5. Demonstração do funcionamento do aplicativo do sistema de fiscalização de veículos

Neste tópico será feita uma breve descrição do funcionamento do aplicativo desenvolvido como protótipo para o Sistema de Fiscalização de Veículos.

Começaremos pela tela inicial do aplicativo. Nesta tela o usuário poderá escolher entre Atualizar Banco de Dados do Sistema de Fiscalização de Veículos ou Efetuar Controle sobre a via de veículos.

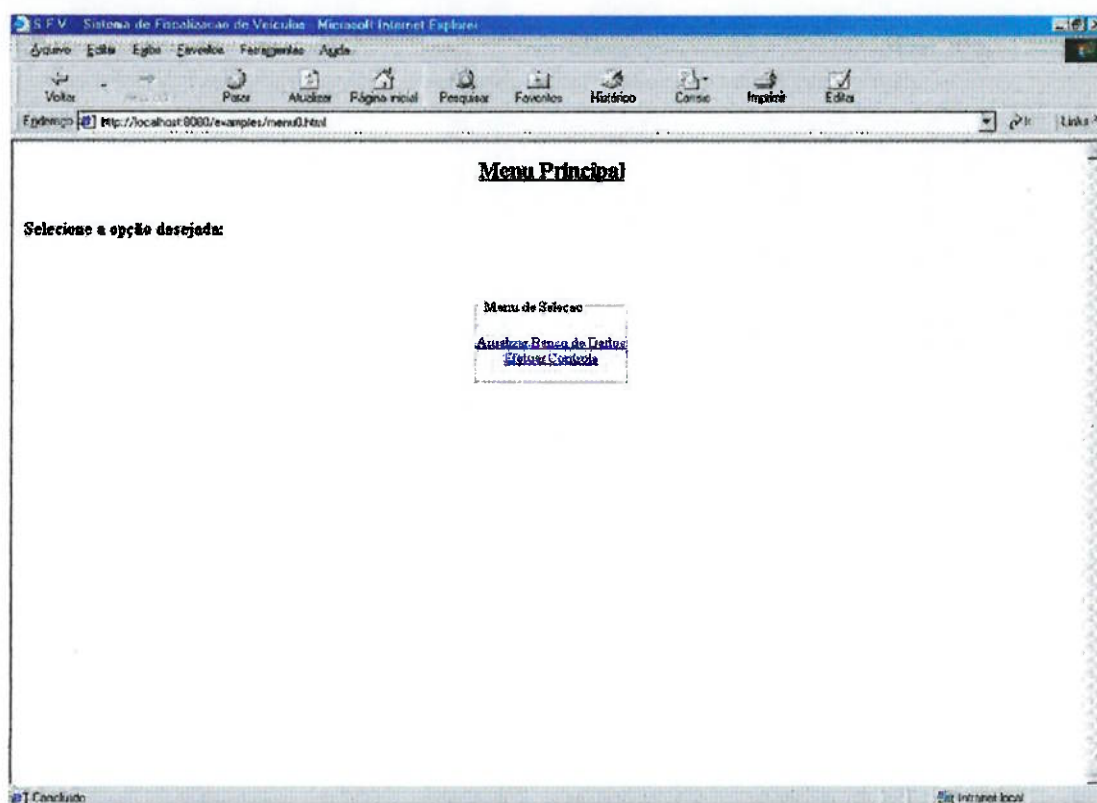


Figura 32: Menu Inicial.

10.5.1. Atualizar o Banco de Dados

Escolhendo a opção de atualizar o banco de dados, ao usuário serão disponibilizadas as seguintes opções: Inserir Veículo, Inserir Local, Consultar Veículo, Consultar Local.

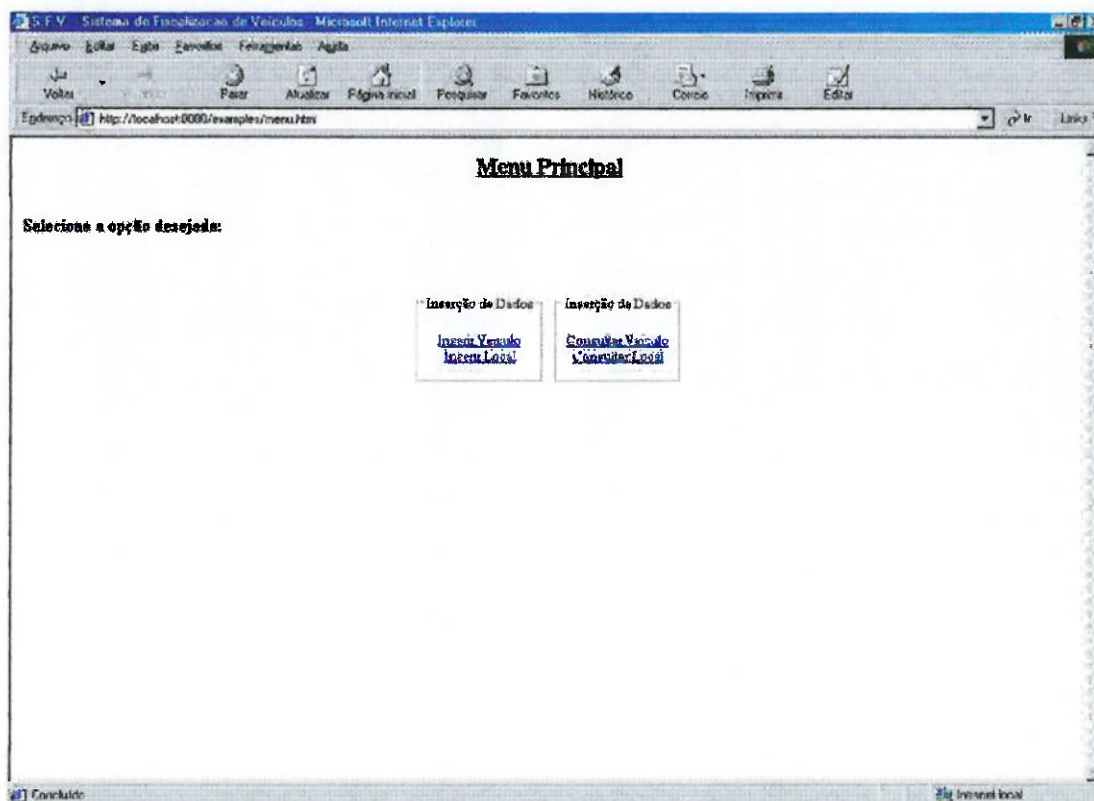


Figura 33: Menu Atualizar Banco de Dados.

10.5.2. Inserir Veículo e Inserir Local

As opções Inserir Veículo e Inserir Local permitem ao usuário a inserção de um registro de veículo ou de um local de fiscalização de veículos no banco de dados do Sistema de Fiscalização de Veículos. Sendo que para cada tela de inserção de dados, dados específicos sobre o novo registro de veículo ou de local de fiscalização de veículos são requisitados.

Após a inserção de um registro no banco de dados, tanto de veículo quanto de local de fiscalização de veículos, uma tela de operação concluída é mostrada para o usuário e o sistema retorna para o menu Atualizar Banco de Dados.

S.F.V. Sistema de Fiscalização de Veículos Microsoft Internet Explorer

Arquivo Editar Exibir Favoritos Ferramentas Ajuda

Voltar Avançar Parar Atualizar Página inicial Pesquisar Favoritos Histórico Conexão Impressão Editar

Endereço http://localhost:8000/exemplos/insere_veiculo.html

Inserção de Veículos no Cadastro

Dados do Veículo

Número do chassi do veículo:	B9G9843MO3ER234
Código do Renavam do veículo:	98573311
Nome do proprietário do veículo:	FLAVIO BERLAND
CPF / CAC do proprietário do veículo:	3254331135
Placa do veículo e UF:	MDA3245
Marca / Modelo do veículo:	KADET
Espécie / Tipo do veículo:	PASSEIO
Ano de fabricação / Ano do modelo do veículo:	1994
Cor predominante do veículo:	PRETO
Veículo Rotulado / Pintado (S/N)	☒ Sim ☐ Não
Documentação Insular (S/N)	☒ Sim ☐ Não

Enviar os dados Apagar os dados

Concluído Microsoft local

Figura 34: Tela de Inserção de Veículos.

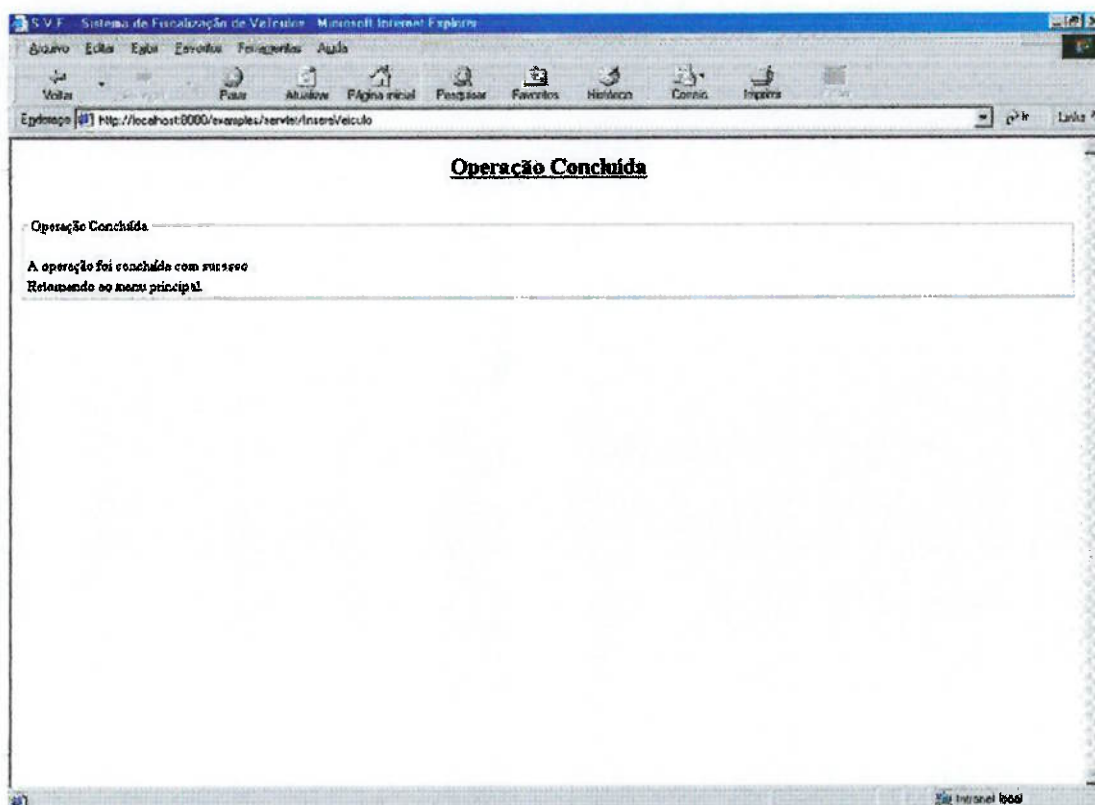


Figura 35: Tela de Operação Concluída.

10.5.3. Consultar Veículo e Consultar Local

As opções Consultar Veículo e Consultar Local permitem ao usuário que este faça uma consulta de um registro de veículo ou de um local de fiscalização de veículos no banco de dados do Sistema de Fiscalização de Veículos. Sendo que, para a tela de consulta de registro de veículos, o dado de entrada pode ser tanto o número do chassi do veículo, o código renavam, ou, a placa do mesmo, e, para a tela de consulta de local de fiscalização de veículos, o dado de entrada pode ser tanto o código de identificação ou o endereço deste local de fiscalização de veículos.

Se nenhum registro, tanto de veículo quanto de local de fiscalização de veículos, foi encontrado o programa mostra um aviso indicando o ocorrido e o usuário tem a opção de retornar à tela Atualizar o Banco de Dados.

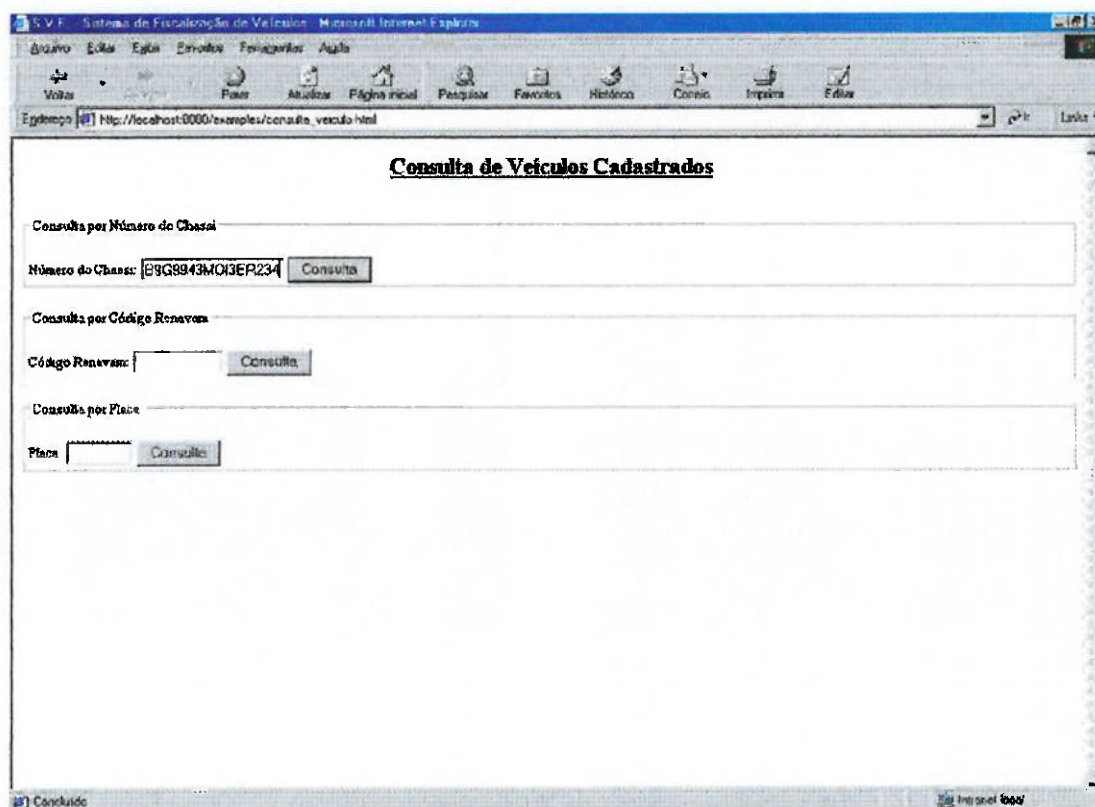


Figura 36: Tela de Consulta de Veículos.

10.5.4. Resultado de Consulta de Veículo ou de Local de Fiscalização de Veículos Cadastrado

Se um registro correspondente a um veículo ou local de fiscalização de veículo cadastrado foi encontrado no banco de dados do Sistema de Fiscalização de Veículos, o programa mostra o conteúdo deste registro na tela e dá as seguintes opções ao usuário: Alterar ou Remover registro.

Se o usuário optar por Remover o registro, ou do veículo ou do local de fiscalização de veículos, este é removido do banco de dados. Em seguida, uma tela de operação concluída é mostrada para o usuário e o sistema retorna para o menu Atualizar Banco de Dados.

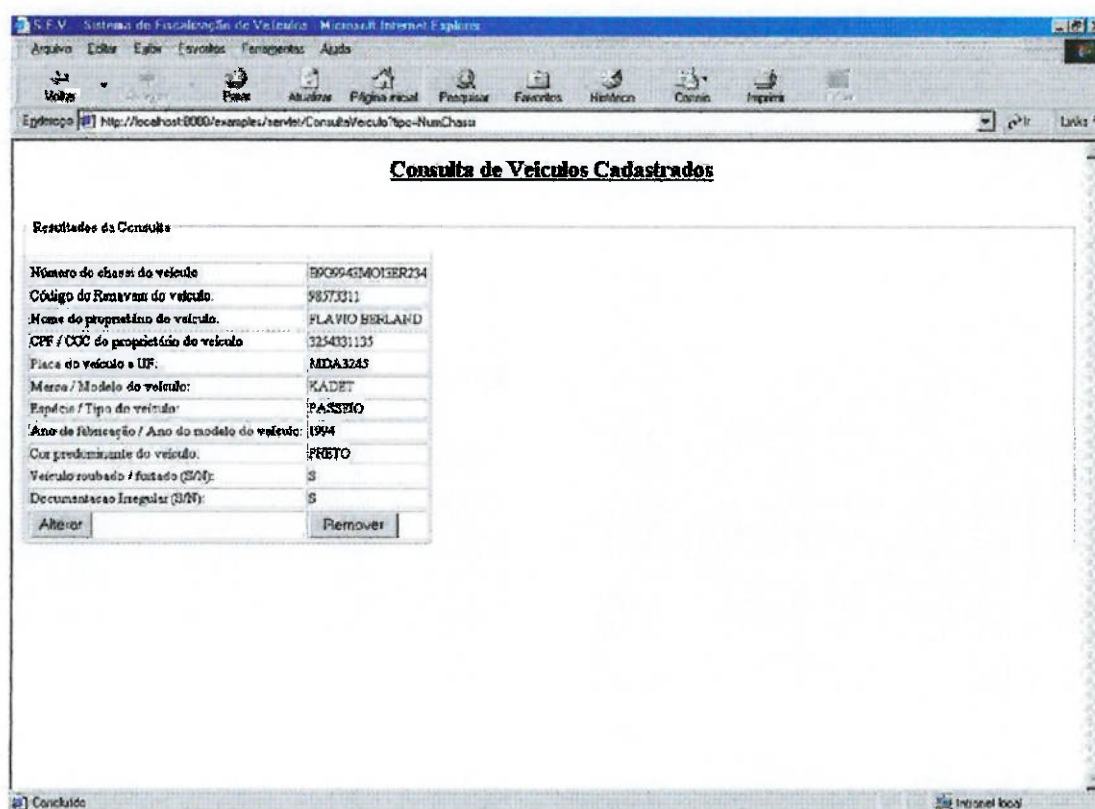


Figura 37: Tela de Resultado de Consulta de Registro de Veículo.

10.5.5. Alteração de Registro de Veículo e Local de Fiscalização de Veículos do Banco de Dados.

Se o usuário optar por Alterar o registro, ou do veículo ou do local de fiscalização de veículos, uma tela de alteração de registro aparece para o mesmo. Esta tela disponibiliza as informações contidas no banco de dados do Sistema de Fiscalização de Veículos para que o usuário possa alterá-las conforme a conveniência desta operação. Em seguida, uma tela de operação concluída é mostrada para o usuário e o sistema retorna para o menu Atualizar Banco de Dados.

S.F.V. Sistema de Fiscalização de Veículos - Microsoft Internet Explorer

Arquivo Editar Exibir Favoritos Ferramentas Ajuda

Voltar Avançar Parar Atualizar Página inicial Pesquisar Favoritos Histórico Correio Imprimir Editar

Endereço: <http://localhost:8000/exemplos/servico/PaginaAlterarVeiculo?NumChassi=E9G9943M012E1224>

Alteração de Veículos Cadastrados

Dados do Veículo

Código do Renavam do veículo: 98573311

Nome do proprietário do veículo: CARLOS OLIVEIRA

CPF / CQC do proprietário do veículo: 3413652465

Placa do veículo e ITP: MQA3245

Marca / Modelo do veículo: KADET

Espécie / Tipo do veículo: PASSEIO

Ano de fabricação / Ano do modelo do veículo: 1994

Cor predominante do veículo: PRETO

Veículo roubado / furtado (S/N): ☐ Sim ☒ Não

Documentação irregular (S/N): ☐ Sim ☒ Não

Concluído

Figura 38: Tela de Alteração de Registro de Veículo.

10.5.6. Efetuar Controle

Nesta tela o Sistema de Fiscalização de Veículos está efetuando o controle sobre a via de circulação de veículos de fato e disponibilizando, quando for o caso, as principais informações sobre os veículos irregulares que circulam sobre esta via.

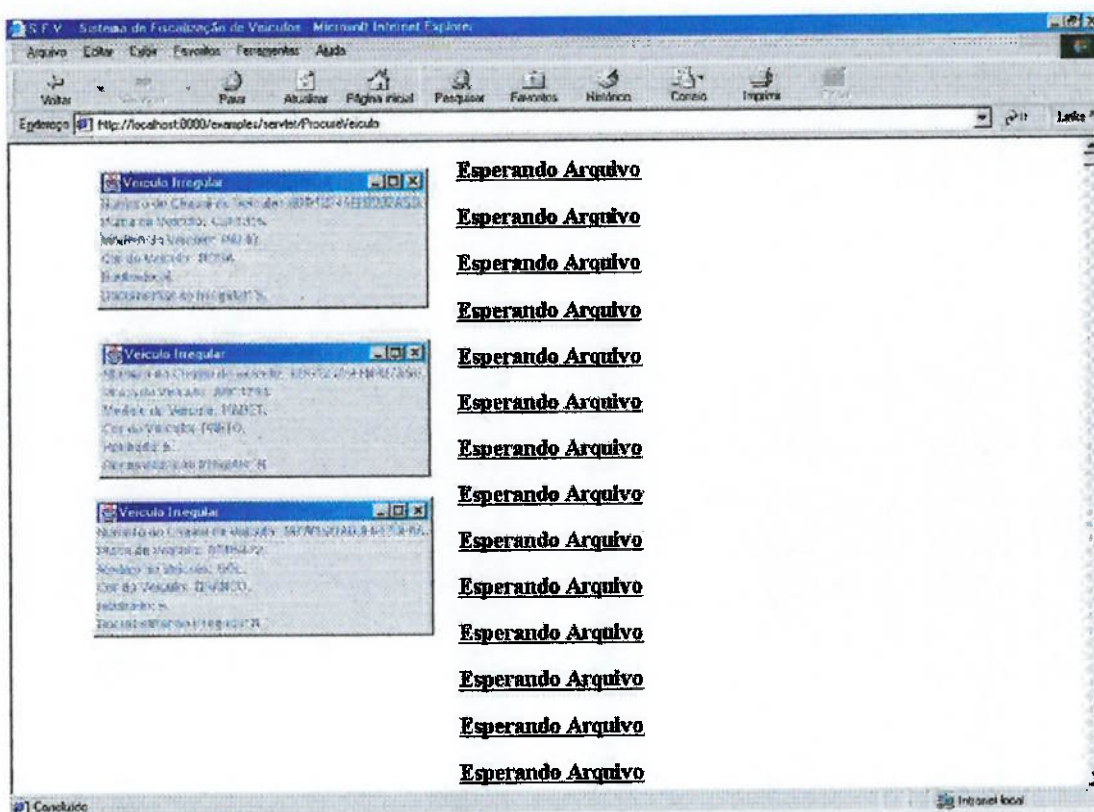


Figura 39: Tela de Efetuação de Controle Sobre uma Via.

11. MÓDULO DE EXIBIÇÃO DE VEÍCULOS IRREGULARES

O módulo para exibição dos veículos irregulares tem como função a exibição de alguns dados do veículo irregular fornecido pela central de dados, de modo que o policial tenha condições de identificar o veículo que deve ser parado.

Para isto, será utilizado um computador portátil equipado com placa de rede Ethernet, padrão 10BaseT. Este laptop estará conectado a um Subscriber Unit do sistema de rádio, de forma que este tenha mobilidade total.

Abaixo, segue um esquema, ilustrando este processo:



Figura 40: Sistema de exibição dos dados dos veículos irregulares.

APÊNDICE A – LISTAGENS

Menu Inicial – menu0.html

```
<HTML>
<HEAD>
  <TITLE>S.F.V. - Sistema de Fiscalizacao de Veiculos</TITLE>
</HEAD>

<BODY>
  <H2><center><u><b>Menu Principal</b></u></center></H2><BR>

  <H3><b>Selecione a opção desejada:</b></H3><BR>
  <br>
  <P ALIGN="CENTER">
    <table>
      <tr>
        <td>
          <FIELDSET><LEGEND ACCESSKEY=A>Menu de Selecao</LEGEND><p>
            <center>
              <A HREF="menu.html">Atualizar Banco de Dados</A><br>
              <A
                HREF="http://localhost:8080/examples/servlet/ProcuraVeiculo">Efetuar
Controle</A><br><br>
            </center>
          </td>
        </tr>
      </table>
    </P>
  </BODY>
</HTML>
```

Menu Atualizar o Banco de Dados – menu.html

```
<HTML>
<HEAD>
  <TITLE>S.F.V. - Sistema de Fiscalizacao de Veiculos</TITLE>
</HEAD>

<BODY>
  <H2><center><u><b>Menu Principal</b></u></center></H2><BR>

  <H3><b>Selecione a opção desejada:</b></H3><BR>
  <br>
  <P ALIGN="CENTER">
    <table>
      <tr>
        <td>
          <FIELDSET><LEGEND ACCESSKEY=A>Inserção de Dados</LEGEND><p>
            <center>
              <A HREF="insere_veiculo.html">Inserir Veiculo</A><br>
              <A HREF="insere_local.html">Inserir Local</A><br><br>
            </center>
          </fieldset>
        </td>
        <td>&nbsp;</td>
        <td>
          <FIELDSET><LEGEND ACCESSKEY=A>Inserção de Dados</LEGEND><p>
            <center>
              <A HREF="consulta_veiculo.html">Consultar Veiculo</A><br>
              <A HREF="consulta_local.html">Consultar Local</A><br><br>
            </center>
          </fieldset>
        </td>
      </tr>
    </table>
  </P>
</BODY>
</HTML>
```

Inserer_Veiculo.html

```
<HTML>
<HEAD>
  <TITLE>S.F.V. - Sistema de Fiscalização de Veículos</TITLE>
</HEAD>

<BODY>
  <H2><center><u><b>Inserção de Veículos no Cadastro </b></u></center></H2><BR>

  <FORM name="InsererVeiculo" ACTION="http://localhost:8080/examples/servlet/InsererVeiculo"
METHOD=POST>
    <FIELDSET><LEGEND ACCESSKEY=A>Dados do Veículo</LEGEND><P>
      <table>
        <tr>
          <td>Número do chassi do veículo:</td>
          <td><INPUT TYPE="text" NAME="NumChassi" SIZE="17"
MAXLENGTH="17"></td>
        </tr>
        <tr>
          <td>Código do Renavam do veículo:</td>
          <td><INPUT TYPE="text" NAME="CodRenavam" SIZE="10"
MAXLENGTH="10"></td>
        </tr>
        <tr>
          <td>Nome do proprietário do veículo:</td>
          <td><INPUT TYPE="text" NAME="NomeProp" SIZE="45"
MAXLENGTH="45"></td>
        </tr>
        <tr>
          <td>CPF / CGC do proprietário do veículo:</td>
          <td><INPUT TYPE="text" NAME="CpfProp" SIZE="12"
MAXLENGTH="12"></td>
        </tr>
        <tr>
          <td>Placa do veículo e UF:</td>
          <td><INPUT TYPE="text" NAME="Placa" SIZE="9"
MAXLENGTH="9"></td>
        </tr>
      </table>
    </P>
  </FORM>
```

```

        <tr>
            <td>Marca / Modelo do veículo:</td>
            <td><INPUT TYPE="text" NAME="Modelo" SIZE="20"
MAXLENGTH="20"></td>
        </tr>
        <tr>
            <td>Espécie / Tipo do veículo:</td>
            <td><INPUT TYPE="text" NAME="TipoVeic" SIZE="20"
MAXLENGTH="20"></td>
        </tr>
        <tr>
            <td>Ano de fabricação / Ano do modelo do veículo:</td>
            <td><INPUT TYPE="text" NAME="AnoFabr" SIZE="9"
MAXLENGTH="9"></td>
        </tr>
        <tr>
            <td>Cor predominante do veículo:</td>
            <td><INPUT TYPE="text" NAME="CorVeic" SIZE="10"
MAXLENGTH="10"></td>
        </tr>
        <tr>
            <td>Veículo Roubado / Furtado (S/N):</td>
            <td><INPUT TYPE="radio" NAME="Roubado" VALUE="S"> Sim
                <INPUT TYPE="radio" NAME="Roubado" VALUE="N" checked>
Nao</td>
        </tr>
        <tr>
            <td>Documentacao Irregular (S/N):</td>
            <td><INPUT TYPE="radio" NAME="DocIrreg" VALUE="S"> Sim
                <INPUT TYPE="radio" NAME="DocIrreg" VALUE="N" checked>
Nao</td>
        </tr>
    </table>
</fieldset>

<br>

<CENTER>

```

```

        <INPUT TYPE="SUBMIT" VALUE="Enviar os dados">
        <INPUT TYPE="RESET" VALUE="Apagar os dados">
    </CENTER>
</FORM>
</BODY>
</HTML>

```

Inserer_Local.html

```

<HTML>
<HEAD>
    <TITLE>S.F.V. - Sistema de Fiscalização de Veículos</TITLE>
</HEAD>

<BODY>
    <H2><center><u><b>Inserção de Informações sobre os Pontos de Fiscalização de
Veículos</b></u></center></H2><BR>

    <FORM name="insere_local" ACTION="http://localhost:8080/examples/servlet/InsereLocal"
METHOD=POST>
        <FIELDSET><LEGEND ACCESSKEY=A>Dados dos Pontos de Fiscalização de
Veículos</LEGEND><P>
            <table>
                <tr>
                    <td>Código de identificação do ponto de fiscalização de veículos:</td>
                    <td><INPUT TYPE="text" NAME="CodFisc" SIZE="10"
MAXLENGTH="10"></td>
                </tr>
                <tr>
                    <td>Local do ponto de fiscalização de veículos:</td>
                    <td><INPUT TYPE="text" NAME="EndFisc" SIZE="45"
MAXLENGTH="45"></td>
                </tr>
            </table>
        </fieldset>

        <br>

        <CENTER>

```



```

        <INPUT TYPE="SUBMIT" VALUE="Enviar os dados">
        <INPUT TYPE="RESET" VALUE="Apagar os dados">
    </CENTER>
</FORM>
</BODY>
</HTML>

```

Consulta_Veiculo.html

```

<HTML>
<HEAD>
    <TITLE>S.V.F. - Sistema de Fiscalização de Veículos</TITLE>
</HEAD>

<BODY>
    <H2><center><u><b>Consulta de Veiculos Cadastrados</b></u></center></H2><BR>

    <form name="consulta_chassi" method="post"
Action="http://localhost:8080/examples/servlet/ConsultaVeiculo?tipo=NumChassi">
        <FIELDSET><LEGEND ACCESSKEY=A>Consulta por Número do
Chassi</LEGEND><P>
            <table>
                <tr>
                    <td>Número do Chassi:</td>
                    <td><INPUT TYPE="text" NAME="valor" SIZE="17"
MAXLENGHT="17"></td>
                    <td><INPUT TYPE="SUBMIT" VALUE="Consulta"></td>
                </tr>
            </table>
        </fieldset>
    </form>

    <form name="consulta_renavam" method="post"
Action="http://localhost:8080/examples/servlet/ConsultaVeiculo?tipo=CodRenavam">
        <FIELDSET><LEGEND ACCESSKEY=A>Consulta por Código
Renavam</LEGEND><P>
            <table>
                <tr>
                    <td>Código Renavam:</td>

```

```

        <td><INPUT TYPE="text" NAME="valor" SIZE="10"
MAXLENGHT="10"></td>
        <td><INPUT TYPE="SUBMIT" VALUE="Consulta"></td>
    </tr>
</table>
</fieldset>
</form>

<form name="consulta_placa" method="post"
Action="http://localhost:8080/examples/servlet/ConsultaVeiculo?tipo=Placa">
    <FIELDSET><LEGEND ACCESSKEY=A>Consulta por Placa</LEGEND><P>
    <table>
        <tr>
            <td>Placa:</td>
            <td><INPUT TYPE="text" NAME="valor" SIZE="7"
MAXLENGHT="7"></td>
            <td><INPUT TYPE="SUBMIT" VALUE="Consulta"></td>
        </tr>
    </table>
</fieldset>
</form>
</BODY>
</HTML>

```

Consulta_Local.html

```

<HTML>
<HEAD>
    <TITLE>S.V.F. - Sistema de Fiscalização de Veículos</TITLE>
</HEAD>

<BODY>
    <H2><center><u><b>Consulta de Locais Cadastrados</b></u></center></H2><BR>

    <form name="consulta_codfisc" method="post"
Action="http://localhost:8080/examples/servlet/ConsultaLocal?tipo=CodFisc">
        <FIELDSET><LEGEND ACCESSKEY=A>Consulta por Código de Identificação do
Local</LEGEND><P>
        <table>

```

```

        <tr>
            <td>Código de Identificação do Local:</td>
            <td><INPUT TYPE="text" NAME="valor" SIZE="20"
MAXLENGHT="20"></td>
            <td><INPUT TYPE="SUBMIT" VALUE="Consulta"></td>
        </tr>
    </table>
</fieldset>
</form>

<form name="consulta_endfisc" method="post"
Action="http://localhost:8080/examples/servlet/ConsultaLocal?tipo=EndFisc">
    <FIELDSET><LEGEND ACCESSKEY=A>Consulta por Local do Ponto de
Fiscalização</LEGEND><P>
        <table>
            <tr>
                <td>Local do Ponto de Fiscalização:</td>
                <td><INPUT TYPE="text" NAME="valor" SIZE="45"
MAXLENGHT="45"></td>
                <td><INPUT TYPE="SUBMIT" VALUE="Consulta"></td>
            </tr>
        </table>
    </fieldset>
</form>

</BODY>
</HTML>

```

InserVeiculo.Java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class InserVeiculo extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("Erro de ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {

```

```

resp.setContentType("text/html");
PrintWriter out = new PrintWriter(resp.getOutputStream());

// get request parameters

// Numero do Chassi do Veiculo
String szNumChassi = "null";
szNumChassi = req.getParameter("NumChassi");

// Codigo Renavam
String szCodRenavam = "null";
szCodRenavam = req.getParameter("CodRenavam");

// Nome do Proprietario
String szNomeProp = "null";
szNomeProp = req.getParameter("NomeProp");

// CPF ou CGC do Proprietario
String szCpfProp = "null";
szCpfProp = req.getParameter("CpfProp");

// Placa do Veiculo e UF
String szPlaca = "null";
szPlaca = req.getParameter("Placa");

// Marca e Modelo do Veiculo
String szModelo = "null";
szModelo = req.getParameter("Modelo");

// Especie e Tipo do Veiculo
String szTipoVeic = "null";
szTipoVeic = req.getParameter("TipoVeic");

// Ano de Fabricação e Modelo
String szAnoFabr = "null";
szAnoFabr = req.getParameter("AnoFabr");

// Cor Predominante

```

```

String szCorVeic = "null";
    szCorVeic = req.getParameter("CorVeic");

    // Veiculo Roubado ou Furtado
String szRoubado = "null";
    szRoubado = req.getParameter("Roubado");

    // Documentacao Irregular
String szDocIrreg = "null";
    szDocIrreg = req.getParameter("DocIrreg");

try {
    PreparedStatement stmt= m_dbConnection.prepareStatement("insert into veiculos values
    (?,?,?,?,?,?,?,?,?,?)");
    stmt.setString(1, szNumChassi.toUpperCase());
    stmt.setString(2, szCodRenavam.toUpperCase());
    stmt.setString(3, szNomeProp.toUpperCase());
    stmt.setString(4, szCpfProp);
    stmt.setString(5, szPlaca.toUpperCase());
    stmt.setString(6, szModelo.toUpperCase());
    stmt.setString(7, szTipoVeic.toUpperCase());
    stmt.setString(8, szAnoFabr.toUpperCase());
    stmt.setString(9, szCorVeic.toUpperCase());
    stmt.setString(10, szRoubado.toUpperCase());
    stmt.setString(11, szDocIrreg.toUpperCase());

    if (stmt.executeUpdate() != 1) {
        out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro durante a execução desta operação.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    } else {

```

```

        out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de
Veículos</TITLE>");
        out.println("<script>");
        out.println("function goToMenu()");
        out.println("{ window.navigate('http://localhost:8080/examples/menu.html'); }");
        out.println("</script></HEAD><BODY>");
        out.println("<script>window.setTimeout('goToMenu()',5000,'JavaScript');</script>");
        out.println("<H2><center><u><b>Operação Concluída</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Operação Concluída</LEGEND><p><table>");
        out.println("<tr><td>A operação foi concluída com sucesso.</td></tr>");
        out.println("<tr><td>Retornando ao menu principal.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
} catch (Exception e) {
    out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
    out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
    out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
    out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
    out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
    out.println("<tr><td>" + e.toString() + "</td></tr>");
    out.println("</table></fieldset><br>");
    out.println("</FORM></BODY></HTML>");
}
    out.close();
} // doPost

} // InserirVeiculo

```

InserirLocalJava

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class InserirLocal extends HttpServlet

```

```

{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("Erro de ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = new PrintWriter(resp.getOutputStream());

        // get request parameters

        // Codigo do local de fiscalizacao
        String szCodFisc = "null";
        szCodFisc = req.getParameter("CodFisc");
    }
}

```



```

        // Endereco do local de fiscalizacao
        String szEndFisc = "null";
        szEndFisc = req.getParameter("EndFisc");

try {
    PreparedStatement stmt= m_dbConnection.prepareStatement("insert into locais values (?,?)");
    stmt.setString(1, szCodFisc.toUpperCase());
    stmt.setString(2, szEndFisc.toUpperCase());

    if (stmt.executeUpdate() != 1) {
        out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de  
Veículos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no  
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro durante a execução desta operação.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    } else {
        out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de  
Veículos</TITLE>");
        out.println("<script>");
        out.println("function goToMenu()");
        out.println("{ window.navigate('http://localhost:8080/examples/menu.html'); }");
        out.println("</script></HEAD><BODY>");
        out.println("<script>>window.setTimeout('goToMenu()',5000,'JavaScript');</script>");
        out.println("<H2><center><u><b>Operação Concluída</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Operação Concluída</LEGEND><p><table>");
        out.println("<tr><td>A operação foi concluída com sucesso.</td></tr>");
        out.println("<tr><td>Retornando ao menu principal.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
} catch (Exception e) {

```

```

        out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
    out.close();
} // doPost

} // InsercLocal

```

ConsultaVeiculo.Java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class ConsultaVeiculo extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);

```

```

        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("error in ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException
    {
        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = new PrintWriter(resp.getOutputStream());

        // get request parameters
        String campo = "null";
        campo = req.getParameter("tipo");
        String valor = "null";
        valor = req.getParameter("valor");

    try {
        Statement stmt = m_dbConnection.createStatement();
        ResultSet rs = stmt.executeQuery("Select * from veiculos where " + campo + " like '" + valor +
        "%'");

        out.println("<HTML>");
        out.println("<HEAD>");
        out.println("<TITLE>S.F.V. - Sistema de Fiscalização de Veículos</TITLE>");
        out.println("</HEAD>");
        out.println("<BODY>");
        out.println("<H2><center><u><b>Consulta de Veiculos  

Cadastrados</b></u></center></H2><BR>");

```

```

        out.println("<FIELDSET><LEGEND ACCESSKEY=A>Resultados da
Consulta</LEGEND><P>");
        out.println("<table border='1'>");

        if(rs.next()) {

            String szNumChassi = "null";
            szNumChassi = rs.getString(1);

            out.println("<tr>");
            out.println("<tr><td>Número do chassi do veículo:</td><td>" + szNumChassi +
"</td></tr>");
            out.println("<tr><td>Código do Renavam do veículo:</td><td>" + rs.getString(2) +
"</td></tr>");
            out.println("<tr><td>Nome do proprietário do veículo:</td><td>" + rs.getString(3) +
"</td></tr>");
            out.println("<tr><td>CPF / CGC do proprietário do veículo:</td><td>" + rs.getString(4) +
"</td></tr>");
            out.println("<tr><td>Placa do veículo e UF:</td><td>" + rs.getString(5) + "</td></tr>");
            out.println("<tr><td>Marca / Modelo do veículo:</td><td>" + rs.getString(6) + "</td></tr>");
            out.println("<tr><td>Espécie / Tipo do veículo:</td><td>" + rs.getString(7) + "</td></tr>");
            out.println("<tr><td>Ano de fabricação / Ano do modelo do veículo:</td><td>" +
rs.getString(8) + "</td></tr>");
            out.println("<tr><td>Cor predominante do veículo:</td><td>" + rs.getString(9) +
"</td></tr>");
            out.println("<tr><td>Veículo roubado / furtado (S/N):</td><td>" + rs.getString(10) +
"</td></tr>");
            out.println("<tr><td>Documentacao Irregular (S/N):</td><td>" + rs.getString(11) +
"</td></tr>");

            String szChangeCommand = "null";
            String szDeleteCommand = "null";
            szChangeCommand =
"javascript:window.navigate('http://localhost:8080/examples/servlet/PaginaAlteraVeiculo?NumChassi
=" + szNumChassi + "')";

```

```

        szDeleteCommand =
"javascript:window.navigate('http://localhost:8080/examples/servlet/DeletaVeiculo?NumChassi=" +
szNumChassi + "');";

        out.println("<td><input type='button' value='Alterar' OnClick=" + szChangeCommand +
"></td>");
        out.println("<td><input type='button' value='Remover' OnClick=" + szDeleteCommand +
"></td>");
        out.println("</tr>");

    } else {
        String szChangeCommando = "null";
        szChangeCommando =
"javascript:window.navigate('http://localhost:8080/examples/menu.html');";
        out.println("<td><input type='button' value='Nenhum Registro Encontrado' OnClick=" +
szChangeCommando + "></td>");
    }
    out.println("</table>");
    out.println("</FIELDSET>");
    out.println("<br>");
    out.println("</BODY>");
    out.println("</HTML>");

    out.println("</table>");
    out.println("</body>");
    out.println("</html>");

} catch (Exception e) {
    out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veiculos</TITLE></HEAD>");
    out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
    out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
    out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
    out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
    out.println("<tr><td>" + e.toString() + "</td></tr>");
    out.println("</table></fieldset><br>");
    out.println("</FORM></BODY></HTML>");
}

```

```

    }

    out.close();
} // doPost

} //ConsultaVeiculo

```

ConsultaLocal.Java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class ConsultaLocal extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("error in ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        doPost(req, resp);
    }

```

```

    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = new PrintWriter(resp.getOutputStream());

        // get request parameters
        String campo = "null";
        campo = req.getParameter("tipo");
        String valor = "null";
        valor = req.getParameter("valor");

        try {
            Statement stmt = m_dbConnection.createStatement();
            ResultSet rs = stmt.executeQuery("Select * from locais where " + campo + " like '" + valor +
            "%'");

            out.println("<HTML>");
            out.println("<HEAD>");
            out.println("<TITLE>S.F.V. - Sistema de Fiscalização de Veículos</TITLE>");
            out.println("</HEAD>");
            out.println("<BODY>");
            out.println("<H2><center><u><b>Consulta de Veiculos  

            Cadastrados</b></u></center></H2><BR>");
            out.println("<FIELDSET><LEGEND ACCESSKEY=A>Resultados da  

            Consulta</LEGEND><P>");
            out.println("<table border='1'>");

            if(rs.next()) {

                String szCodFisc = "null";
                szCodFisc = rs.getString(1);

                out.println("<tr>");

```

```

        out.println("<tr><td>Código de Identificação do Local:</td><td>" + szCodFisc +
"</td></tr>");
        out.println("<tr><td>Local do Ponto de Fiscalização:</td><td>" + rs.getString(2) +
"</td></tr>");

        String szChangeCommand = "null";
        String szDeleteCommand = "null";
        szChangeCommand =
"javascript:window.navigate('http://localhost:8080/examples/servlet/PaginaAlteraLocal?CodFisc=" +
szCodFisc + "');";
        szDeleteCommand =
"javascript:window.navigate('http://localhost:8080/examples/servlet/DeletaLocal?CodFisc=" +
szCodFisc + "');";

        out.println("<td><input type='button' value='Alterar' onClick=" + szChangeCommand +
"></td>");
        out.println("<td><input type='button' value='Remover' onClick=" + szDeleteCommand +
"></td>");
        out.println("</tr>");

    } else {
        String szChangeCommando = "null";
        szChangeCommando =
"javascript:window.navigate('http://localhost:8080/examples/menu.html');";
        out.println("<td><input type='button' value='Nenhum Registro Encontrado' onClick=" +
szChangeCommando + "></td>");
    }
    out.println("</table>");
    out.println("</FIELDSET>");
    out.println("<br>");
    out.println("</BODY>");
    out.println("</HTML>");

    out.println("</table>");
    out.println("</body>");
    out.println("</html>");

} catch (Exception e) {

```



```

        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veiculos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados. </td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema. </td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
    out.close();
    } // doPost

} //ConsultaLocal

```

Deleta Veiculo.Java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class DeletaVeiculo extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);

```

```

        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("Erro de ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException
    {
        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = new PrintWriter(resp.getOutputStream());

        // get request parameters
        // NumChassi
        String szNumChassi = "null";
        szNumChassi = req.getParameter("NumChassi");

        try {
            PreparedStatement stmt= m_dbConnection.prepareStatement("delete from veiculos
where NumChassi=?");
            stmt.setString(1, szNumChassi);

            if (stmt.executeUpdate() != 1) {
                out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
                out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
                out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
                out.println("<tr><td>Ocorreu um erro durante a execução desta operação.</td></tr>");
                out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
                out.println("</table></fieldset><br>");
                out.println("</FORM></BODY></HTML>");
            } else {

```

```

        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE>");
        out.println("<script>");
        out.println("function goToMenu()");
        out.println("{ window.navigate('http://localhost:8080/examples/menu.html'); }");
        out.println("</script></HEAD><BODY>");
        out.println("<script>>window.setTimeout('goToMenu()',5000,'JavaScript');</script>");
        out.println("<H2><center><u><b>Operação Concluída</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Operação Concluída</LEGEND><p><table>");
        out.println("<tr><td>A operação foi concluída com sucesso.</td></tr>");
        out.println("<tr><td>Retornando ao menu principal.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
} catch (Exception e) {
    out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
    out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
    out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
    out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
    out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
    out.println("<tr><td>" + e.toString() + "</td></tr>");
    out.println("</table></fieldset><br>");
    out.println("</FORM></BODY></HTML>");
    }
    out.close();
}
} //DeletaVeiculo

```

DeletaLocal.Java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class DeletaLocal extends HttpServlet
{

```

```

Connection m_dbConnection = null;
static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
static final String DSN = "jdbc:odbc:sfv";
static final String USER = "jm";
static final String PWD = "jm";

public void init(ServletConfig config) throws ServletException
{
    super.init(config);

    // open database connection
    try {
        Class.forName(ODBC_DRIVER).newInstance();
        m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
    } catch (Exception e) {
        e.printStackTrace();
        throw new ServletException("Erro de ODBC");
    }
} // init

public void doGet(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
{
    doPost(req, resp);
} // doGet

public void doPost(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
{
    resp.setContentType("text/html");
    PrintWriter out = new PrintWriter(resp.getOutputStream());

    // get request parameters
    // NumChassi
    String szCodFisc = "null";
    szCodFisc = req.getParameter("CodFisc");
    try {

```

```

        PreparedStatement stmt= m_dbConnection.prepareStatement("delete from locais where
CodFisc=?");

        stmt.setString(1, szCodFisc);

        if (stmt.executeUpdate() != 1) {
            out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
            out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
            out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
            out.println("<tr><td>Ocorreu um erro durante a execução desta operação.</td></tr>");
            out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
            out.println("</table></fieldset><br>");
            out.println("</FORM></BODY></HTML>");
        } else {
            out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE>");
            out.println("<script>");
            out.println("function goToMenu()");
            out.println("{ window.navigate('http://localhost:8080/examples/menu.html'); }");
            out.println("</script></HEAD><BODY>");
            out.println("<script>window.setTimeout('goToMenu()',5000,'JavaScript');</script>");
            out.println("<H2><center><u><b>Operação Concluída</b></u></center></H2><BR>");
            out.println("<fieldset><LEGEND>Operação Concluída</LEGEND><p><table>");
            out.println("<tr><td>A operação foi concluída com sucesso.</td></tr>");
            out.println("<tr><td>Retornando ao menu principal.</td></tr>");
            out.println("</table></fieldset><br>");
            out.println("</FORM></BODY></HTML>");
        }
    } catch (Exception e) {
        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
    }
}

```

```

        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
    out.close();
}
} //DeletaLocal

```

AlteraVeiculo.Java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class AlteraVeiculo extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("Erro de ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {

```

```

        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = new PrintWriter(resp.getOutputStream());

        // get request parameters

        // Numero do Chassi do Veiculo
        String szNumChassi = "null";
        szNumChassi = req.getParameter("NumChassi");

        // Codigo Renavam
        String szCodRenavam = "null";
        szCodRenavam = req.getParameter("CodRenavam");

        // Nome do Proprietario
        String szNomeProp = "null";
        szNomeProp = req.getParameter("NomeProp");

        // CPF ou CGC do Proprietario
        String szCpfProp = "null";
        szCpfProp = req.getParameter("CpfProp");

        // Placa do Veiculo e UF
        String szPlaca = "null";
        szPlaca = req.getParameter("Placa");

        // Marca e Modelo do Veiculo
        String szModelo = "null";
        szModelo = req.getParameter("Modelo");

        // Especie e Tipo do Veiculo
        String szTipoVeic = "null";
        szTipoVeic = req.getParameter("TipoVeic");
    }

```

```

        // Ano de Fabricação e Modelo
        String szAnoFabr = "null";
        szAnoFabr = req.getParameter("AnoFabr");

        // Cor Predominante
        String szCorVeic = "null";
        szCorVeic = req.getParameter("CorVeic");

        // Veiculo Roubado ou Furtado
        String szRoubado = "null";
        szRoubado = req.getParameter("Roubado");

        // Documentacao Irregular
        String szDocIrreg = "null";
        szDocIrreg = req.getParameter("DocIrreg");

    try {
        PreparedStatement stmt= m_dbConnection.prepareStatement("update veiculos set
        CodRenavam=?,NomeProp=?,CpfProp=?,Placa=?,Modelo=?,TipoVeic=?,AnoFabr=?,CorVeic=?,Rou
        bado=?,DocIrreg=? where NumChassi=?");
        stmt.setString(1, szCodRenavam.toUpperCase());
        stmt.setString(2, szNomeProp.toUpperCase());
        stmt.setString(3, szCpfProp);
        stmt.setString(4, szPlaca.toUpperCase());
        stmt.setString(5, szModelo.toUpperCase());
        stmt.setString(6, szTipoVeic.toUpperCase());
        stmt.setString(7, szAnoFabr.toUpperCase());
        stmt.setString(8, szCorVeic.toUpperCase());
        stmt.setString(9, szRoubado.toUpperCase());
        stmt.setString(10, szDocIrreg.toUpperCase());
        stmt.setString(11, szNumChassi);

        if (stmt.executeUpdate() != 1) {
            out.println("<HTML><HEAD><TITLE>S. V.F. - Sistema de Fiscalização de
            Veículos</TITLE></HEAD>");
            out.println("<BODY><H2><center><u><b>Erro no
            Sistema</b></u></center></H2><BR>");

```



```

        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro durante a execução desta operação.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    } else {
        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE>");
        out.println("<script>");
        out.println("function goToMenu()");
        out.println("{ window.navigate('http://localhost:8080/examples/menu.html'); }");
        out.println("</script></HEAD><BODY>");
        out.println("<script>>window.setTimeout('goToMenu()',5000,'JavaScript');</script>");
        out.println("<H2><center><u><b>Operação Concluída</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Operação Concluída</LEGEND><p><table>");
        out.println("<tr><td>A operação foi concluída com sucesso.</td></tr>");
        out.println("<tr><td>Retornando ao menu principal.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
} catch (Exception e) {
    out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
    out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
    out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
    out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
    out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
    out.println("<tr><td>" + e.toString() + "</td></tr>");
    out.println("</table></fieldset><br>");
    out.println("</FORM></BODY></HTML>");
}

    out.close();
} // doPost

} // AlteraVeiculo

```

AlterLocal.Java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class AlterLocal extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {
        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("Erro de ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
    {
        resp.setContentType("text/html");
```

```

        PrintWriter out = new PrintWriter(resp.getOutputStream());

        // get request parameters

        // Numero do Chassi do Veiculo
        String szCodFisc = "null";
        szCodFisc = req.getParameter("CodFisc");

        // Codigo Renavam
        String szEndFisc = "null";
        szEndFisc = req.getParameter("EndFisc");

        try {
            PreparedStatement stmt= m_dbConnection.prepareStatement("update locais set EndFisc=?
where CodFisc=?");
            stmt.setString(1, szEndFisc.toUpperCase());
            stmt.setString(2, szCodFisc);

            if (stmt.executeUpdate() != 1) {
                out.println("<HTML><HEAD><TITLE>S.V.F. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
                out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
                out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
                out.println("<tr><td>Ocorreu um erro durante a execução desta operação.</td></tr>");
                out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
                out.println("</table></fieldset><br>");
                out.println("</FORM></BODY></HTML>");
            } else {
                out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE>");
                out.println("<script>");
                out.println("function goToMenu()");
                out.println("{ window.navigate('http://localhost:8080/examples/menu.html'); }");
                out.println("</script></HEAD><BODY>");
                out.println("<script>window.setTimeout('goToMenu()',5000,'JavaScript');</script>");
                out.println("<H2><center><u><b>Operação Concluída</b></u></center></H2><BR>");
                out.println("<fieldset><LEGEND>Operação Concluída</LEGEND><p><table>");
            }
        }
    }

```

```

        out.println("<tr><td>A operação foi concluída com sucesso.</td></tr>");
        out.println("<tr><td>Retornando ao menu principal.</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
    } catch (Exception e) {
        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veículos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
    out.close();
    } // doPost

} // AlteraLocal

```

PaginaAlterarVeiculo.java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class PaginaAlterarVeiculo extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";
    static final String USER = "jm";
    static final String PWD = "jm";

    public void init(ServletConfig config) throws ServletException
    {

```

```

        super.init(config);

        // open database connection
        try {
            Class.forName(ODBC_DRIVER).newInstance();
            m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException("error in ODBC");
        }
    } // init

    public void doGet(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException
    {
        doPost(req, resp);
    } // doGet

    public void doPost(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException
    {
        resp.setContentType("text/html");
        PrintWriter out = new PrintWriter(resp.getOutputStream());

        String szNumChassi = "null";
        szNumChassi = req.getParameter("NumChassi");

        try {
            Statement stmt = m_dbConnection.createStatement();
            ResultSet rs = stmt.executeQuery("select * from veiculos where NumChassi like '" +
            szNumChassi + "'");

            rs.next();

            out.println("<HTML>");
            out.println("<HEAD>");
            out.println("<TITLE>S.F.V. - Sistema de Fiscalização de Veiculos</TITLE>");

```

```

        out.println("</HEAD>");
        out.println("<BODY>");
        out.println("<H2><center><u><b>Alteração de Veículos
Cadastrados</b></u></center></H2><BR>");
        out.println("<FORM name='altera_veiculo'
ACTION='http://localhost:8080/examples/servlet/AlteraVeiculo?NumChassi=' + szNumChassi + '"
METHOD=POST>");
        out.println("<FIELDSET><LEGEND ACCESSKEY=A>Dados do Veículo</LEGEND><P>");
        out.println("<table>");

        out.println("<tr>");
        out.println("<td>Código do Renavam do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='CodRenavam' SIZE='10' MAXLENGTH='10'
value='" + rs.getString(2) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Nome do proprietário do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='NomeProp' SIZE='45' MAXLENGTH='45'
value='" + rs.getString(3) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>CPF / CGC do proprietário do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='CpfProp' SIZE='12' MAXLENGTH='12'
value='" + rs.getString(4) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Placa do veículo e UF:</td>");
        out.println("<td><INPUT TYPE='text' NAME='Placa' SIZE='9' MAXLENGTH='9'
value='" + rs.getString(5) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Marca / Modelo do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='Modelo' SIZE='20' MAXLENGTH='20'
value='" + rs.getString(6) + "'></td>");

```

```

        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Espécie / Tipo do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='TipoVeic' SIZE='20' MAXLENGTH='20'
value='" + rs.getString(7) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Ano de fabricação / Ano do modelo do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='AnoFabr' SIZE='9' MAXLENGTH='9'
value='" + rs.getString(8) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Cor predominante do veículo:</td>");
        out.println("<td><INPUT TYPE='text' NAME='CorVeic' SIZE='10' MAXLENGTH='10'
value='" + rs.getString(9) + "'></td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Veículo roubado / furtado (S/N):</td>");
        out.println("<td><INPUT TYPE='radio' NAME='Roubado' VALUE='S'> Sim<INPUT
TYPE='radio' NAME='Roubado' VALUE='N' checked> Nao</td>");
        out.println("</tr>");

        out.println("<tr>");
        out.println("<td>Documentacao Irregular (S/N):</td>");
        out.println("<td><INPUT TYPE='radio' NAME='DocIrreg' VALUE='S'> Sim<INPUT
TYPE='radio' NAME='DocIrreg' VALUE='N' checked> Nao</td>");
        out.println("</tr>");

        rs.close();

        out.println("</td>");
        out.println("</tr>");
        out.println("</table>");
        out.println("</FIELDSET>");

```

```

        out.println("<br>");
        out.println("<CENTER>");
            out.println("<INPUT TYPE='SUBMIT' VALUE='Enviar os dados'>");
            out.println("<INPUT TYPE='RESET' VALUE='Apagar os dados'>");
        out.println("</CENTER>");
        out.println("</FORM>");
        out.println("</BODY>");
        out.println("</HTML>");

        out.println("</table>");
        out.println("</body>");
        out.println("</html>");

    } catch (Exception e) {
        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalizacao de
Veiculos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }

    out.close();
    } // doPost

} // PaginaAlteraVeiculo

```

PaginaAlteraLocal.Java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class PaginaAlteraLocal extends HttpServlet
{

```



```

Connection m_dbConnection = null;
static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
static final String DSN = "jdbc:odbc:sfv";
static final String USER = "jm";
static final String PWD = "jm";

public void init(ServletConfig config) throws ServletException
{
    super.init(config);

    // open database connection
    try {
        Class.forName(ODBC_DRIVER).newInstance();
        m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
    } catch (Exception e) {
        e.printStackTrace();
        throw new ServletException("error in ODBC");
    }
} // init

public void doGet(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
{
    doPost(req, resp);
} // doGet

public void doPost(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
{
    resp.setContentType("text/html");
    PrintWriter out = new PrintWriter(resp.getOutputStream());

    String szCodFisc = "null";
    szCodFisc = req.getParameter("CodFisc");

    try {
        Statement stmt = m_dbConnection.createStatement();

```

```

ResultSet rs = stmt.executeQuery("select * from locais where CodFisc like '" + szCodFisc +
    """);

    rs.next();

    out.println("<HTML>");
    out.println("<HEAD>");
    out.println("<TITLE>S.F.V. - Sistema de Fiscalização de Veiculos</TITLE>");
    out.println("</HEAD>");
    out.println("<BODY>");
    out.println("<H2><center><u><b>Alteração de Locais de Fiscalizacao  
Cadastrados</b></u></center></H2><BR>");
    out.println("<FORM name='altera_veiculo'  
ACTION='http://localhost:8080/examples/servlet/AlterLocal?CodFisc=' + szCodFisc + "  
METHOD=POST>");
    out.println("<FIELDSET><LEGEND ACCESSKEY=A>Dados do Veículo</LEGEND><P>");
    out.println("<table>");

    out.println("<tr>");
    out.println("<td>Local do Ponto de Fiscalização:</td>");
    out.println("<td><INPUT TYPE='text' NAME='EndFisc' SIZE='45' MAXLENGHT='45'  
value='" + rs.getString(2) + "'></td>");
    out.println("</tr>");

    rs.close();

    out.println("</td>");
    out.println("</tr>");
    out.println("</table>");
    out.println("</FIELDSET>");
    out.println("<br>");
    out.println("<CENTER>");
    out.println("<INPUT TYPE='SUBMIT' VALUE='Enviar os dados'>");
    out.println("<INPUT TYPE='RESET' VALUE='Apagar os dados'>");
    out.println("</CENTER>");
    out.println("</FORM>");
    out.println("</BODY>");
    out.println("</HTML>");

```

```

        out.println("</table>");
        out.println("</body>");
        out.println("</html>");

    } catch (Exception e) {
        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalizacao de
Veiculos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }

    out.close();
} // doPost

} // PaginaAlterarLocal

```

ProcuraVeiculo.Java

```

import java.lang.*;
import java.io.*;
import java.io.File.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import javax.swing.*;
import java.awt.*;
import java.util.Arrays;

public class ProcuraVeiculo extends HttpServlet
{
    Connection m_dbConnection = null;
    static final String ODBC_DRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";
    static final String DSN = "jdbc:odbc:sfv";

```

```

static final String USER = "jm";
static final String PWD = "jm";

public void init(ServletConfig config) throws ServletException
{
    super.init(config);

    // open database connection
    try {
        Class.forName(ODBC_DRIVER).newInstance();
        m_dbConnection = DriverManager.getConnection(DSN, USER, PWD);
    } catch (Exception e) {
        e.printStackTrace();
        throw new ServletException("error in ODBC");
    }
} // init

public void doGet(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
{
    doPost(req, resp);
} // doGet

public void doPost(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
{
    PrintWriter out = new PrintWriter(resp.getOutputStream());
    String pathName = "c:\\reccbc\\";
    File arquivo = new File(pathName);
    File theFileName[] = new File[10];
    int inicializa = 0;
    while (true)
    {
        // cria um array de arquivos
        theFileName = arquivo.listFiles();
        // cria um array com nome de arquivos
        String listaDeArquivos[] = new String[10];
        listaDeArquivos = arquivo.list();
    }
}

```

```

// verifica a quantidade de arquivos presentes
int quantidadeDeArquivos = listaDeArquivos.length;
int i = 0;
// enquanto houver arquivos
while (i < quantidadeDeArquivos)
{
    // inicia a leitura dos arquivos
    java.io.BufferedReader theStream = new java.io.BufferedReader(new
java.io.FileReader(theFileName[i]));
    String theInputString = new String();
    theInputString = theStream.readLine();
    // enquanto este arquivo contiver registros
    while (theInputString != null)
    {
        try {
            Statement stmt = m_dbConnection.createStatement();
            // procura por registro de veiculo irregular
            ResultSet rs = stmt.executeQuery("Select Placa, Modelo, CorVeic, Roubado,
DocIrreg from veiculos where NumChassi like '" + theInputString + "%' and ((Roubado like 'S') or
(DocIrreg like 'S')) ");
            // se encontrou o registro de veiculo irregular
            if (rs.next())
            {
                String szPlaca = rs.getString(1);
                String szModelo = rs.getString(2);
                String szCorVeic = rs.getString(3);
                String szRoubado = rs.getString(4);
                String szDocIrreg = rs.getString(5);

                // mostra na tela

                JFrame frame = new JFrame("Veiculo Irregular");

                JLabel NumChassiLabel, PlacaLabel, ModeloLabel, CorLabel, RoubadoLabel,
DocIrregLabel;

                JPanel MostraPanel = new JPanel();
                MostraPanel.setLayout(new GridLayout(6, 1));

```

```

        NumChassiLabel = new JLabel("Numero do Chassi do Veiculo: " +
theInputString + ".");
        PlacaLabel = new JLabel("Placa do Veiculo: " + szPlaca + ".");
        ModeloLabel = new JLabel("Modelo do Veiculo: " + szModelo + ".");
        CorLabel = new JLabel("Cor do Veiculo: " + szCorVeic + ".");
        RoubadoLabel = new JLabel("Roubado: " + szRoubado + ".");
        DocIrregLabel = new JLabel("Documentacao Irregular: " + szDocIrreg + ".");

        MostraPanel.add(NumChassiLabel);
        MostraPanel.add(PlacaLabel);
        MostraPanel.add(ModeloLabel);
        MostraPanel.add(CorLabel);
        MostraPanel.add(RoubadoLabel);
        MostraPanel.add(DocIrregLabel);

        frame.getContentPane().add(MostraPanel);

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.pack();
        frame.setVisible(true);
    }
    // passa para a proxima linha
    rs.close();
    theInputString = theStream.readLine();
    } catch (Exception e) {
        out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veiculos</TITLE></HEAD>");
        out.println("<BODY><H2><center><u><b>Erro no
Sistema</b></u></center></H2><BR>");
        out.println("<fieldset><LEGEND>Erro no Sistema</LEGEND><p><table>");
        out.println("<tr><td>Ocorreu um erro de acesso ao banco de dados.</td></tr>");
        out.println("<tr><td>Entre em contato com o administrador do sistema.</td></tr>");
        out.println("<tr><td>" + e.toString() + "</td></tr>");
        out.println("</table></fieldset><br>");
        out.println("</FORM></BODY></HTML>");
    }
} // while (theInputString != null)

```

```

        // apaga arquivo
        try
        {
            theStream.close();
            theFileName[i].delete();
        }
        catch (Exception e) {
            out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veiculos</TITLE></HEAD>");
            out.println("<BODY><H2><center><u><b>Erro no Sistema - Nao Consegue Apagar
Arquivo</b></u></center></H2></BODY></HTML>");
        }
        // passa para o proximo arquivo
        i = i + 1;
    } // while (i < quantidadeDeArquivos)
    out.println("<HTML><HEAD><TITLE>S.F.V. - Sistema de Fiscalização de
Veiculos</TITLE></HEAD>");
    out.println("<BODY><H2><center><u><b>Esperando
Arquivo</b></u></center></H2></BODY></HTML>");
    } // while (true)
    // out.close();
} // doPost
} // Procura Veiculo

```

APÊNDICE B – GRAUS DE PROTEÇÃO

Generalidades

Costuma-se definir o grau de proteção dos invólucros para equipamentos elétricos mediante uma dupla classificação. A primeira tratando da penetração de corpos estranhos sólidos como poeira e outros e a segunda contra penetração de líquidos como água e outros.

Os graus de proteção são definidos mediante as letras IP seguidas de duas cifras. A primeira cifra caracteriza o grau de proteção de pessoas contra os contatos com as partes sob tensão ou peças em movimento dentro dos invólucros e o grau de proteção do material contra penetração de corpos sólidos estranhos. A segunda cifra caracteriza o grau de proteção contra penetração de líquidos.

Norma

Descrição detalhada dos graus de proteção e dos ensaios necessários encontram-se nas normas EB 1017 e IEC 529.

Primeira Cifra Característica

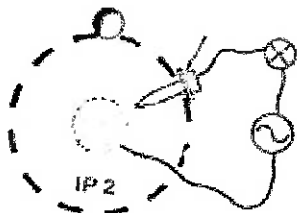


IP0x - Sem Proteção

Ensaio exigido: Nenhum

IP1x - Protegido contra os corpos sólidos superiores a 50mm.

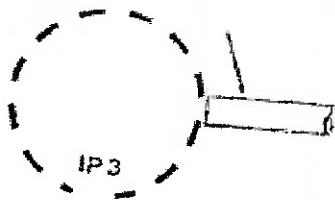
Ensaio exigido: Com esfera diâmetro 52,5mm



IP2x - Protegido contra os corpos sólidos superiores a 12mm

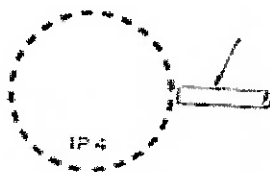
Ensaio exigido: 1) Com esfera diâmetro 12,5mm

2) Com dedo de prova



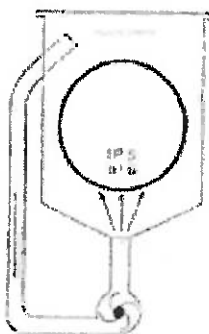
IP3x - Protegido contra corpos sólidos superiores a 2,5mm

Ensaio exigido: Com fio de aço de diametro 2,5mm



IP4x - Protegido contra corpos sólidos superiores a 1mm

Ensaio exigido: Com fio de aço de diametro 1mm



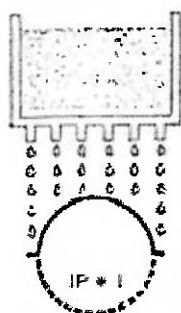
IP5x - Protegido contra poeira, porem a poeira pode penetrar mas não em quantidade suficiente que possa prejudicar a operação do equipamento

Ensaio exigido: Vide desenho.

IP6x - Totalmente protegido contra poeira.

Ensaio exigido: Mesmo procedimento de teste do IP5x.

Segunda Cifra Característica

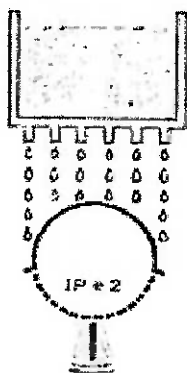


IPx0 - Sem proteção

Ensaio exigido : Nenhum

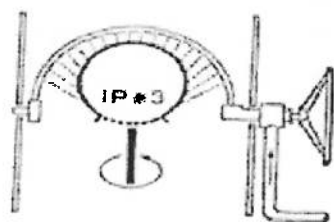
IPx1 - Protegido contra quedas verticais de gotas de água

Ensaio exigido: Coloca-se o equipamento abaixo de um tanque com água contendo abertura que permitem a descida de água em ângulo reto, tempo de teste 10 min.



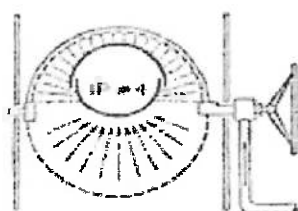
IPx2 - Protegido contra quedas de gotas de água até 15° da vertical

Ensaio exigido: Coloca-se o equipamento abaixo de um tanque com água contendo abertura que permitem a descida de água em ângulo reto movimenta-se o equipamento + - 15°, tempo de teste 10 min.



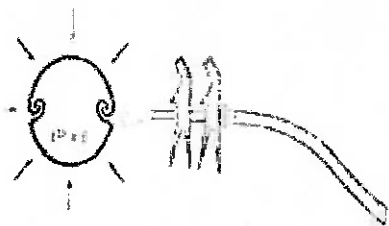
Px3 - Protegido contra água pluvial até 60° da vertical

Ensaio exigido : Coloca-se o equipamento abaixo de uma mangueira colocada entre dois pilares (semi circulo) com furos em ângulo de 60° da vertical e movimenta-se o equipamento, tempo de teste 10 min.



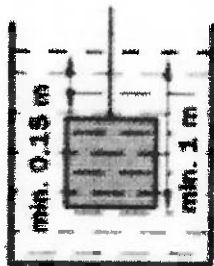
IPx4 - Protegido contra projeções de água em qualquer direção

Ensaio exigido: Projeta-se água contra o equipamento em todas as direções (Ex. Chuva com vento), tempo de teste 1min.



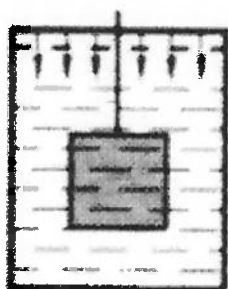
IPx5 - Protegido contra impacto de ondas marítimas ou jatos potentes

Ensaio exigido: Projeta-se água no equipamento com mangueiras de incêndio, durante 1 min.



IPx6 - Protegido contra imersão.

Ensaio exigido: Durante 1 min. conforme desenho



IPx7 - Protegido contra submersão.

Ensaio exigido: câmara fechada com água, durante 30min.

Níveis de Proteção

1ª cifra característica	0	1	2	3	4	5	6	7
0	IP 00	IP 01	IP 02					
1	IP 10	IP 11	IP 12	IP 13				
2	IP 20	IP 21	IP 22	IP 23				
3	IP 30	IP 31	IP 32	IP 33	IP 34			
4	IP 40	IP 41	IP 42	IP 43	IP 44	IP 45	IP 46	
5					IP 54	IP 55	IP 56	
6						IP 65	IP 66	IP 67

BIBLIOGRAFIA

Balogh. Available at: <http://www.balogh-usa.com> [20/10/2002]

Conjuntura Criminal. Available at: <http://www.conjunturacriminal.com.br/> [20/06/2002].

Campbell, Patrick T., Instalando redes em pequenas e médias empresas, São Paulo: Makron Books 1997.

License Plate Recognition – A Tutorial. Available at: <http://www.licenseplaterecognition.com/> [20/06/2002].

Secretaria da Segurança Pública de São Paulo. Available at: <http://www.ssp.sp.gov.br/> [20/06/2002].

Siemens. Catálogos de produtos diversos.

Site da Sun. Available at: <http://www.sun.com> [02/09/2002].

Vehicle Number Plate Recognition Systems. Available at: <http://www.utad.pt/numberplate/> [20/06/2002].