

Centro Universitário Positivo - UnicenP
Núcleo de Ciências Exatas e Tecnológicas – NCET
Engenharia da Computação
Egon Robert Hübner

SISTEMA DE COBRANÇA NON-STOP PARA VEÍCULOS AUTOMOTORES
VIA RFID

Curitiba

2005

Centro Universitário Positivo - UnicenP
Núcleo de Ciências Exatas e Tecnológicas – NCET
Engenharia da Computação
Egon Robert Hübner

SISTEMA DE COBRANÇA NON-STOP PARA VEÍCULOS AUTOMOTORES
VIA RFID

Monografia apresentada à disciplina de Projeto Final, como requisito parcial à conclusão do Curso de Engenharia da Computação.

Orientador: Prof. Marcelo Mikosz Gonçalves

Curitiba

2005

SUMÁRIO

Sumário	II
Lista de Tabelas e Quadros	IV
Lista de Ilustrações.....	IV
Lista de Siglas	V
Resumo.....	VI
Abstract	VI
Introdução.....	VII
1 Aspectos gerais.....	8
1.1 Tema.....	8
1.2 Problema.....	9
1.3 Objetivos	9
1.4 Motivação.....	9
2 Especificação.....	11
2.1 Conceitos Gerais do Projeto	11
2.2 Hardware	13
2.2.1 Funções.....	13
2.2.2 Diagrama em blocos.....	13
2.2.3 Forma de Comunicação.....	14
2.2.3.1 EIA RS-232	14
2.2.3.2 EIA RS-485	14
2.2.3.3 Interface PC ⇔ RFID	16
2.2.3.4 Interface comunicador RFID ⇔ Antena de leitura/gravação	17
2.2.3.5 Interface comunicador RFID ⇔ Sensores	18
2.2.3.6 Interface PC ⇔ WebCam	19
2.2.3.7 Interface PC ⇔ Celular C55 Siemens.....	19
2.3 Software	20
2.3.1 Funções.....	20
2.3.2 Diagrama em blocos.....	20
2.3.3 Ferramentas utilizadas no projeto	21
2.4 Validação de Hardware e Software	22
2.5 Estudo de Viabilidade Técnico-Econômica	22
2.6 Cronograma.....	23
3 projeto de hardware.....	24
3.1 Lógica de funcionamento	24
3.2 Entradas e Saídas digitais.....	24
3.3 Nível lógico de entradas e Saídas digitais	24
3.4 Materiais utilizados	26
4 projeto de Software	27
4.1 Protocolo EUROX V4 - formato da mensagem.....	28
4.1.1 Resumo de caracteres especiais.....	28
4.1.2 Exemplo de transmissão PC ==> Comunicador RFID	28
4.2 Documentos.....	29
4.2.1 Programação orientada a objetos:	29
4.2.1.1 Lógica.....	29
4.2.1.2 Diagramas de casos de uso.....	30
4.2.1.2.1 Cobrança eletrônica via RFID.....	30

4.2.1.2.2	Cadastro de Data-Tag's	31
4.2.1.2.3	Cadastro de Veículos.....	32
4.2.1.2.4	Cadastro de Clientes/Proprietários	33
4.2.1.3	Diagrama de classes	34
4.2.1.4	Diagramas de seqüência	35
4.2.1.4.1	Cobrança.....	35
4.2.1.4.2	Cadastro Proprietário.....	36
4.2.1.4.3	Cadastro Data-Tags	37
4.2.1.4.4	Cadastro de Veículos.....	38
4.2.1.5	Dados.....	39
4.2.1.5.1	Modelo Relacional	39
4.2.1.5.2	Dicionário de Dados.....	40
	Resultados	42
	Conclusão	43
	Anexos.....	44
	Anexo 01 – Distância máxima entre data tag e antena de leitura	44
	Anexo 02 – Conector de Entradas e Saídas Digitais	44
	Anexo 03 – Script SQL de Criação da Base de Dados	45
	Referências Bibliográficas	52

LISTA DE TABELAS E QUADROS

Tabela 1 - Comparação entre os Padrões RS-232 e RS-422	14
Tabela 2 - Características do Padrão RS-485	15
Tabela 3 – Configuração Comunicador PC3141_03B - Jumper 3: Taxa de Baudrate [bps]	17
Quadro 01 – Descrição de Entradas e Saídas Digitais da unidade central RFID	24
Quadro 02 – Descrição Materiais e Equipamentos utilizados no Projeto	26

LISTA DE ILUSTRAÇÕES

Figura 1 – Representação do Projeto em diagrama tecnológico	12
Figura 2 – Representação do Hardware em diagrama de blocos	13
Figura 3 – Representação do conector de interface serial (RS232 & RS485)	16
Figura 4 – Representação do conector da antena de leitura/gravação	18
Figura 5 – Representação do Software em diagrama de blocos	20
Figura 6 – Tecnologias de Software previstas para o projeto	21
Figura 6a – Tecnologias de Software efetivamente utilizadas no projeto	21
Figura 7 – Modelo Entidade-Relacionamento	39
Figura 8 – Representação da distância de leitura/escrita da antena 3114/00B	44
Figura 9 – Representação do conector de E/S digitais do comunicador RFID	44

LISTA DE SIGLAS

EDI – *Electronic Data Interchange* - intercâmbio eletrônico de dados

E/S – Entradas / Saídas

NCET – Núcleo de Ciências Exatas e Tecnológicas

PC – *Personal Computer / Computador Desktop*

RFID – Sistema de Identificação por Rádio Frequência

SMS – *Short Messaging System* – Mensagens de texto para telefonia celular

UNICENP – Centro Universitário Positivo

RESUMO

Implementação de um sistema cobrança eletrônica para veículos automotores em movimento, através de sistema de RFID, apresentado para o curso de Graduação em Engenharia da Computação, no Centro Universitário Positivo.

A demanda por serviços rápidos de tráfego, a fim de diminuir tempos de espera para os motoristas bem como diminuição de filas, melhoria do fluxo e redução de custos para os empreendedores, tem levado várias empresas de tráfego e donos de estacionamentos a buscarem uma alternativa ao sistema tradicional de pagamentos.

A solução apresentou-se através de sistemas de cobrança integrados aos bancos, por intermédio de serviços pós-pagos, com débito em conta corrente ou via cartão de crédito.

Pensando em uma alternativa diferente, propõe-se no presente projeto uma solução mais avançada, com sistema de créditos atualizado no próprio tag regravável, evitando os incômodos de clientes com contas não pagas e a burocracia de contestações de pagamento indevido em função do furto e/ou clonagem de data tag's.

Palavras-chave: Sistema de pagamento integrado via RFID; cobrança eletrônica para veículos automotores; Pedágio *NonStop*

ABSTRACT

This work presents the development of a computer-based system designed to charge vehicles in movement through parking lots and/or toll gates.

The demand for fast services, reduced queues and waiting times for car drivers allied with reduced costs and the total control of transactions, has been a challenge for parking lots and toll gates.

The system has as serial RFID (Radio Frequency ID) interface for communication with a personal computer, two antennas and a lot of transponders, which stores the user data.

Keywords: Integrated RFID charge System, vehicular electronic charge system, NonStop toll gates.

INTRODUÇÃO

A atual tendência de pagamentos integrados automaticamente sem interação humana, a diminuição de filas e a minimização do tempo de espera dos clientes traz consigo uma demanda por serviços rápidos e tecnologias de aplicação recente no comércio. Paralelamente com o advento dessas tecnologias, existe a busca por sistemas cada vez mais seguros e confiáveis, com a utilização de mecanismos de rastreabilidade, redundância e criptografia, principalmente em se tratando de transações financeiras.

Através do presente trabalho, objetiva-se fornecer não só uma solução técnica para integração de sistemas de RFID em âmbito de automação comercial, mas sim prover uma solução prática e economicamente viável para sistemas de tráfego em envolvidos com o pagamento de tarifas para veículos automotores em movimento.

1 ASPECTOS GERAIS

1.1 TEMA

O tema principal do projeto é o desenvolvimento de um sistema automatizado de pagamento de tarifas de pedágio e estacionamento, com cobrança eletrônica através do controle de créditos armazenados em um Data Tag (transponder) afixado no vidro interno de cada veículo.

Quando o veículo em movimento passar pelo corredor de pedágio ou na saída do estacionamento, um sensor indutivo irá disparar o processo de cobrança, descontando os créditos diretamente no data-tag do veículo. Através da leitura do tag, o sistema também atualizará os créditos em uma base de dados, devendo checar a consistência das informações do veículo, em relação ao cadastro no banco de dados. Após a passagem do veículo, o sistema deve enviar uma mensagem de SMS e e-mail ao proprietário do veículo, contendo os créditos atuais, notificação de dia e horário, bem como uma foto digital do veículo no instante em que foi registrada a passagem pela antena de leitura/gravação.

O sistema de RFID consiste de uma antena conectada a uma interface de leitura/escrita (comunicador), que por sua vez possui uma interface serial para fazer a comunicação com um PC dedicado.

O projeto de hardware consiste em criar um multiplexador para a interface de comunicação (comunicador), possibilitando a utilização de duas antenas conectadas através de uma única interface de comunicação serial.

Para possibilitar a automatização do processo de leitura, serão conectados alguns sensores indutivos à interface RFID, os quais deverão ser configurados via jumpers na unidade de RFID, e a sua ativação será programada via interface serial.

Adicionalmente será incorporada uma *WebCam* ao PC dedicado, para envio de imagens ao proprietário do veículo, recurso ativado a cada evento de passagem por um sistema de cobrança RFID.

O projeto de software consiste basicamente de 3 módulos:

- Módulo RFID: comunicação serial através de protocolo EUROX entre comunicador RFID e PC dedicado; comandos de formatação/leitura/gravação de dados através de acesso à estrutura de arquivos do *data tag*; comandos de acesso às entradas/saídas do comunicador RFID; envio de imagens *WebCam*;
- Módulo Email/SMS: envio dos dados das cobranças (transações) da base de dados para o proprietário do veículo, através de celular e e-mail.
- Módulo Cliente Web: informações on-line (base de dados, *WebCam*, RFID).

1.2 PROBLEMA

Realizar a cobrança eletrônica de tarifas de pedágio ou estacionamentos, com o veículo em movimento, sem a necessidade de interação do motorista com um operador, possibilitando um fluxo de velocidade constante na rodovia e na saída de estacionamentos.

1.3 OBJETIVOS

O objetivo principal é o desenvolvimento de um sistema ágil para cobrança de pedágio e tarifas de estacionamento, sem a necessidade de interação humana e sem a necessidade imperiosa da diminuição de velocidade do veículo, por ocasião da passagem em um posto de pedágio ou saída de estacionamento.

Adicionalmente o sistema deve possibilitar um sistema simples de rastreabilidade integrada de veículos, auxiliando os proprietários de veículos a identificar veículos possivelmente clonados, através do envio de mensagem e foto digital com a notificação da passagem nos postos de cobrança.

A integração da base de dados com um sistema de identificação central (DENATRAN, Polícia Federal) não está no escopo do projeto.

1.4 MOTIVAÇÃO

A idéia inicial para o projeto surgiu com base na observação de limitações existentes no controle de tráfego existente atualmente em postos de pedágio, nas filas de pagamento em estacionamentos de aeroportos e postos de combustível. Decidiu-se pela automação de sistemas de pedágio e estacionamentos de aeroportos ou shopping centers, pois envolve poucas empresas, geralmente de grande porte. Com isso elimina-se a necessidade de um complexo sistema integrado de intercâmbio eletrônico de dados (EDI). Caso fosse feita a opção por postos de combustível, haveria a necessidade de se cadastrar dezenas de estabelecimentos, com sistemas e arquiteturas totalmente diversos, em muitos casos deficientes, precários e de manutenção inexistente.

Paralelamente com a idéia de diminuição de filas e minimização do tempo de espera dos clientes surgiu a idéia de se utilizar no projeto algumas tecnologias de aplicação recente no comércio. Utilizado há vários anos em processos de fabricação seriais, os transponders de RF lentamente vêm despertando interesse nos operadores logísticos e em algumas cadeias de comércio. Com o barateamento dos data-tag's, grandes redes de distribuição e supermercados estão despertando para as possibilidades de sistemas de RFID.

Tendo acesso ao empréstimo de equipamentos industriais de RFID, veio a iniciativa de se utilizar um desses equipamentos em algum projeto de automação comercial/corporativo.

Aliado à idéia de fugir do ambiente de programação industrial, base de ocupação na vida profissional cotidiana, procurou-se aproveitar o ensejo do projeto final para utilizar-se de ferramentas bastante atuais não vistas durante o curso. Portanto colocou-se como desafio pessoal, aprender e utilizar mecanismos de rastreabilidade, autenticidade, criptografia, XML em sistemas distribuídos, Web forms, Servlets, Java Server Pages e outras ferramentas atuais do mercado que não constam do currículo e ementa do curso.

2 ESPECIFICAÇÃO

2.1 CONCEITOS GERAIS DO PROJETO

Para facilitar o entendimento da interação Hardware↔Software, torna-se relevante apresentar uma visão do projeto como um todo, demonstrando tanto interfaces físicas quanto lógicas.

O projeto foi concebido para operar em dois modos distintos: modo automático - orientado a eventos externos (sensores de presença veicular) ou modo manual – com envio de comandos por solicitação do usuário.

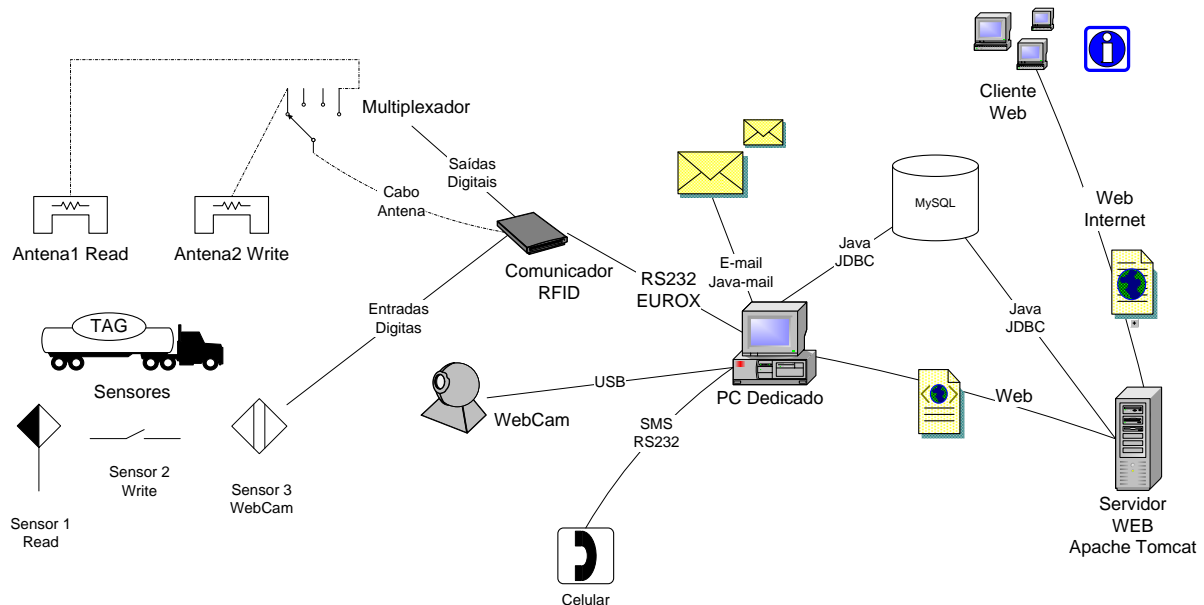
No modo de cobrança automático, quando a *thread* de leitura das entradas digitais detecta a presença de um veículo através do primeiro sensor de presença conectado ao comunicador de RFID, ocorre um processo de leitura do *data-tag*, por intermédio da primeira antena. Faz-se então a leitura do código serial do datatag (ID), data da bateria, número de créditos, código de autenticação (BCC – *block check code*) e da categoria do veículo. Em seguida, desencadeia-se um processo de autenticação do veículo e atualização de créditos, com base nos dados lidos. Terminados os cálculos, o PC envia os novos créditos para um buffer de memória na interface do comunicador. Caso os créditos tenham sido suficientes, aciona-se os LED's verdes, simbolizando a abertura da cancela. De modo contrário, caso não haviam créditos suficientes, acendem-se os LED's vermelhos, indicando que o motorista terá de desviar para a central de atendimento, para recarregar os créditos. Quando o veículo passa no segundo sensor, o PC já finalizou o processamento e os dados de créditos e autenticação já estão disponíveis na memória da interface, para imediatamente serem gravados no Tag. Quando o veículo passa no terceiro sensor, aciona-se a *webcam* para registrar uma foto do veículo. Neste momento, os dados da transação são persistidos na base e os faróis (LED's de sinalização) são novamente apagados, indicando o início de um novo ciclo de cobrança. Uma aplicação paralela, executa uma *thread* cíclica verificando o status do flag de envio de e-mail. Quando uma nova cobrança é registrada na base de dados, essa *thread* se conecta ao servidor de e-mail, enviando os dados da transação ao dono do veículo, contendo informações atualizadas sobre créditos disponíveis, data, hora e imagem da última passagem por um posto de cobrança. Da mesma forma opera um terceiro aplicativo distribuído, o qual acessando a base de dados verifica quais proprietários estão com os créditos terminando, e quando restar apenas uma cobrança, envia uma mensagem SMS ao celular do proprietário.

Paralelamente à descrição anterior, onde o sistema opera disparado por eventos externos, pode ocorrer a operação manual, por exemplo em uma operação de recarga de créditos. Assim, quando algum funcionário desejar recarregar os créditos de um veículo na central de atendimento, utiliza a função de recarga do software, momento em que o data-tag

será atualizado com a quantidade de créditos previamente inserida através de um formulário *Web*.

Toda visualização de dados de veículos, proprietários, data-tags e cobranças poderá ser feita a partir de terminais *Web*, com acesso ao *WebServer*. Neste caso, o servidor *Web* se responsabiliza por buscar ou atualizar as informações na base de dados.

Figura 1 – Representação do Projeto em diagrama tecnológico



A figura acima demonstra a inter-relação entre as tecnologias de hardware e software, de forma ilustrativa. Observa-se que à esquerda do PC dedicado aparecem todos componentes de hardware, como sendo interface de comunicação RFID (comandos de leitura e escrita dos data-tags), *WebCam* (captação de imagens), interface serial para celular (envio de mensagens de texto SMS) e modem (envio de *e-mail*). Do lado direito, aparecem os elementos vinculados intrinsecamente ao software como o sistema gerenciador de banco de dados (MySQL) e o servidor de páginas JSP e Servlets (Apache TomCat).

Um dos conceitos fundamentais do projeto é a divisão dos módulos em sistemas distribuídos, podendo por exemplo os módulos de envio de SMS e *E-mail* serem executados em qualquer máquina ao longo da rede, precisando apenas ter acesso à base de dados.

Da mesma forma, o servidor de páginas pode estar em qualquer máquina distinta do PC dedicado, desde que tenha acesso à base de dados. Especificamente na solução adotada, é necessária uma conexão direta ao PC dedicado, pois o mesmo armazena as imagens captadas nas transações de cobrança. Portanto toda interface com o usuário ocorre em um ambiente gráfico, de uso intuitivo e fácil aprendizagem.

2.2 HARDWARE

2.2.1 Funções

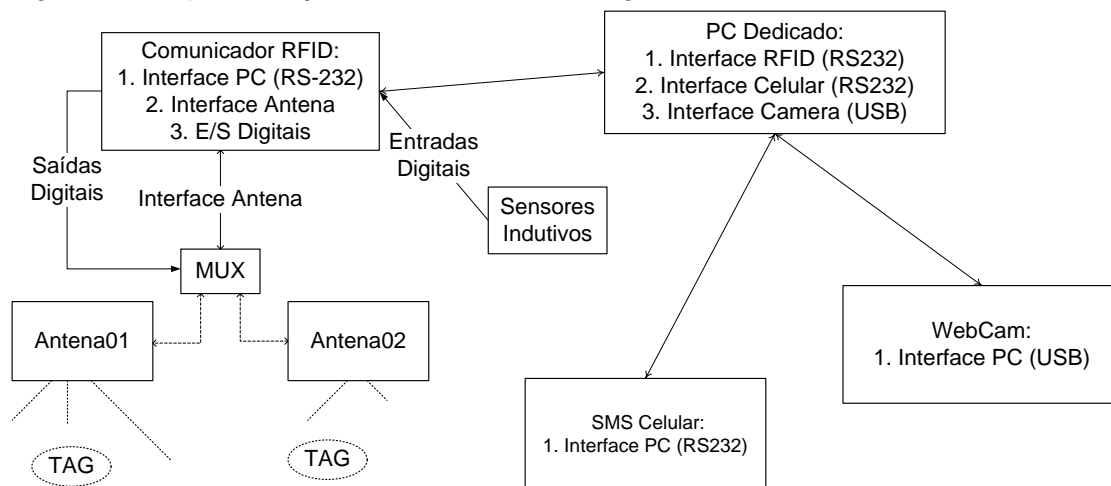
O hardware utilizado no presente projeto consiste de uma interface de comunicação RFID da empresa Baumer Ident, modelo PC3141_03B, duas antenas de leitura/escrita, 2 *transponders* regraváveis de 32kByte, 1 sensor indutivo, 2 sensores mecânicos, 1 multiplexador para duas antenas de RF, uma *webcam* modelo LG LIC300, um telefone Celular modelo Siemens C55 com cabo de comunicação serial, conectores e cabos diversos. O hardware tem a função de atualizar os créditos monetários nos *data tag's* dos veículos, bem como disponibilizar e armazenar informações sobre os veículos.

2.2.2 Diagrama em blocos

Conforme mencionado anteriormente, o hardware original não dispõe da capacidade de acionar mais do que uma antena de leitura e escrita. Para realizar essa função, foi desenvolvido um conjunto multiplexador, cuja lógica combinacional é selecionada via saídas digitais do comunicador de RFID. Essas saídas por sua vez, são acionadas através de comandos Eurox programados remotamente via lógica de software.

Portanto, para o comunicador, as antenas adicionais multiplexadas são totalmente transparentes, comportando-se como se houvesse apenas uma unidade instalada, para não interferir no processo de funcionamento normal.

Figura 2 – Representação do Hardware em diagrama de blocos



A ilustração acima demonstra as várias interfaces físicas, bem como os periféricos envolvidos no projeto. Conforme visto nos conceitos gerais do projeto, o PC dedicado coordena a leitura dos sensores de veículos instalados no piso (função dos laços indutivos em um projeto comercial), comanda o acionamento dos LED's de sinalização (função da cancela automática em um projeto comercial), controla o multiplexador de antenas, através de intertravamentos de software, envia e recebe comandos da unidade de comunicação

RFID, envia SMS via interface de comunicação com celular, capta as imagens dos veículos através de câmera *Web* e envia mensagens de e-mail aos proprietários de veículos, através de modem instalado no PC de envio.

2.2.3 Forma de Comunicação

A comunicação da unidade de interface RFID (comunicador) com o PC dedicado é efetuada através de interface serial, padrão EIA RS-232, para manter a compatibilidade com a interface padrão PC. Na prática, aconselha-se utilizar o padrão de tensão RS-485, em função das distâncias entre o comunicador e o PC, sendo então necessário adequar o sinal de tensão.

Muitos dispositivos utilizados em aplicações industriais utilizam os padrões EIA RS-232, RS-422 ou RS-485 para a comunicação entre dispositivos ou processadores. Erroneamente tem-se o conceito de que estes padrões definem protocolos de comunicação específicos. Os padrões ANSI/EIA RS-xxx especificam apenas as características elétricas.

2.2.3.1 EIA RS-232

O padrão RS-232, também referenciado como interface CCITT V.24, é uma conexão serial encontrada tipicamente em PC's. É utilizado para diversos propósitos como conexão de mouse, impressora, modem, bem como instrumentação industrial. Porém este padrão é limitado à conexão ponto-a-ponto entre a porta serial do PC e o dispositivo.

Tabela 1 - Comparação entre os Padrões RS-232 e RS-422

Parâmetros	EIA RS-232	EIA RS-422
Taxa de transmissão	19200 bps (max.)	10 Mbps (max.)
Distância de transmissão	15 m (max.)	1200 m (max.)
Processo	Desbalanceado	Diferencial
Transmissores	1	1
Receptores	1	10
Princípio	Full-duplex, ponto-a-ponto	Full-duplex, ponto-a-ponto

Fonte: NATIONAL INSTRUMENTS (Tradução do Inglês)

2.2.3.2 EIA RS-485

O padrão RS-485 de interface serial fornece conectividade a vários dispositivos industriais, de manufatura e de aquisição de dados laboratoriais. Em aplicações industriais, a grande maioria dos dispositivos interligados, tais como CLP's, periféricos IHM, *drives* de acionamento e módulos E/S *single-point* utilizam RS-485 para longas distâncias e capacidade multi-ponto.

“The noise immunity and multidrop capability make RS-485 the serial connection of choice in industrial applications requiring many distributed devices networked to a PC or other controller for data collection, HMI, or other operations.”¹

RS-485 é o padrão de comunicação bidirecional mais utilizado em aplicações industriais, e a grande maioria dos barramentos de campo, utiliza-o em sua camada física. Possui transmissão balanceada e suporta conexões multi-ponto, o que permite a criação de redes com até 32 nós e transmissão à distância de até 1200m. Através da inserção de repetidores RS-485 pode-se estender a distância de transmissão em mais 1200m e adicionar outros 32 módulos. Este padrão suporta comunicação *half-duplex*, requer apenas 2 fios para a transmissão e recepção dos dados e possui alta imunidade a ruído.

A interface RS-485 apresenta tensão balanceada semelhante ao RS-422-A, mas permite múltiplos transmissores e receptores operando no mesmo barramento, composto por dois fios. Tipicamente, o padrão RS-485 permite um comprimento máximo de 1200m de cabo entre os equipamentos e uma taxa máxima de 10Mbps de transmissão de dados.

Uma outra vantagem adicional que o RS-485 apresenta sobre o RS-422-A é que ele é capaz de aceitar tensões de modo comum de +12V a -7V, e produz menor quantidade de ruídos. Tensão de modo comum é definida como sendo a média aritmética entre as tensões nos dois terminais do receptor, com relação ao terra.

Tabela 2 - Características do Padrão RS-485

Parâmetros	EIA RS-485
Taxa de transmissão	10 Mbps (máx.)
Distância de transmissão	1200 m (máx.)
Processo	Diferencial (balanceado)
Transmissores	32
Receptores	32
Princípio	Half-duplex, multidrop
Tensão de saída do transmissor	1,5 V (min.)
Resistência de carga	60 ohms (min.)
Máxima corrente de saída em curto-circuito	150mA (p/ o terra) - 250mA (p/ -8V ou -12V)
Resistência de saída em alta impedância	120 kohms (on/off)
Sensibilidade do receptor	200 mV
Resistência de entrada do receptor	12 kohms
Tensão máxima de modo comum	+12 V a -7 V

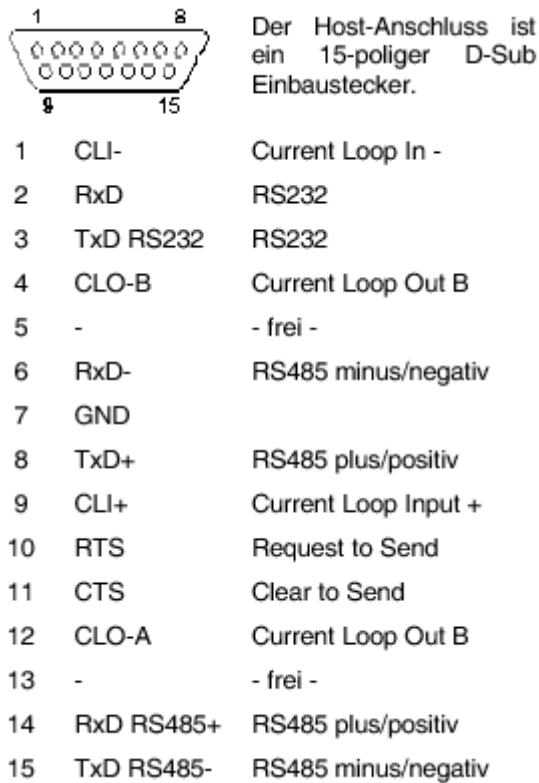
Fonte: NATIONAL INSTRUMENTS (Tradução do Inglês)

¹ “A imunidade a ruídos e a capacidade multi-ponto fazem de RS-485 a escolha certa em conexão serial para aplicações industriais, que requerem vários dispositivos distribuídos conectados a PC’s ou outros controladores para a aquisição de dados, interfaces homem-máquina ou outras operações.” NATIONAL INSTRUMENTS. *What is Serial?* [O que é Serial?] Internet Site <http://www.ni.com>, 2001.

2.2.3.3 Interface PC ⇔ RFID

Para realizar a conexão entre o comunicador RFID e o PC dedicado no padrão RS-232, torna-se necessário conhecer a pinagem do conector da interface de RFID. Como mencionado anteriormente, o comunicador oferece a possibilidade de se conectar em vários níveis de tensão serial, e para tanto utiliza-se um conector DB15, com a seguinte configuração:

Figura 3 – Representação do conector de interface serial (RS232 & RS485)



Fonte: BAUMER IDENT

Na prática, confeccionou-se um cabo blindado com um conector DB15 fêmea do lado do comunicador (interface Host RFID) e um conector DB9 fêmea do lado do PC (porta serial RS-232), ficando o conector com a seguinte pinagem:

Cabo Host - comunicação serial - RS232

DB15	DB9
pino 2 (verde)	pino 3 RxD
pino 3 (azul)	pino 2 TxD
pino 7 (rosa)	pino 5 GND
pino 10 (amarelo)	pino 8 CTS
pino 11 (cinza)	pino 7 RTS
	pino 4 e 6 em ponte (opcional)

Para adequar a velocidade de transmissão da comunicação serial, o comunicador RFID possui jumpers que podem ser adaptados conforme tabela a seguir:

Tabela 3 – Configuração Comunicador PC3141_03B - Jumper 3: Taxa de Baudrate [bps]

Baud	SW3:8	SW3:7	SW3:6	
150	Aus	Aus	Aus	
300	Aus	Aus	Ein	
600	Aus	Ein	Aus	
1200	Aus	Ein	Ein	
2400	Ein	Aus	Aus	
4800	Ein	Aus	Ein	
9600	Ein	Ein	Aus	Standard
19200	Ein	Ein	Ein	

Fonte : BAUMER IDENT (tradução do autor: Ein = Ligado; Aus = Desligado)

No projeto utilizou-se a comunicação em 19200 BAUD, para garantir a máxima velocidade no envio de dados. Essa configuração de velocidade é obtida com os 3 jumpers em posição ligada.

2.2.3.4 Interface comunicador RFID ⇔ Antena de leitura/gravação

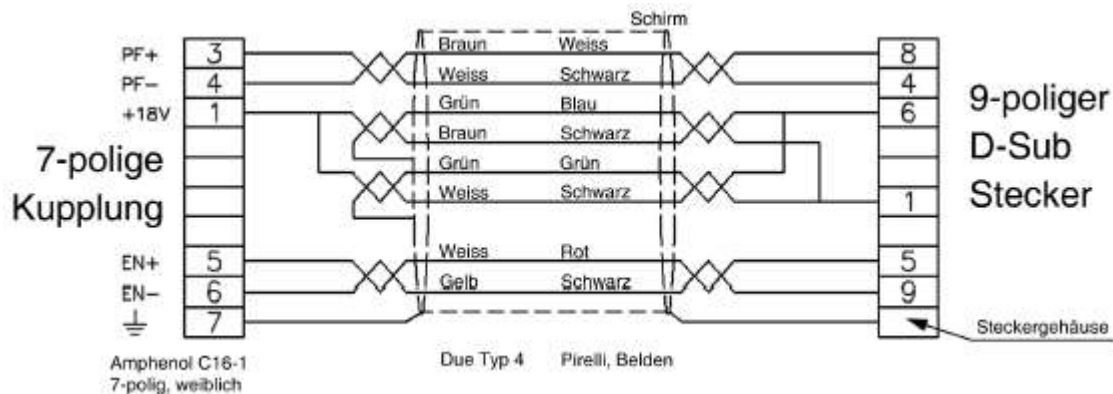
O modelo de comunicador utilizado (Baumer Ident - PC3141_03) originalmente apenas suporta uma antena conectada ao mesmo. O desafio de hardware do presente projeto consistiu em multiplexar as antenas, de modo a possibilitar a utilização de duas antenas conectadas apenas a uma central, diminuindo de forma drástica os custos do projeto e possibilitando separar os processos de leitura e escrita, e com isso ganhar tempo de processamento do PC.

Para realizar a multiplexação das antenas, realizou-se o estudo de chaveadores de alta frequência, pois o comunicador e as antenas operam em uma frequência de 2,45GHz. Na prática verificou-se que os relés de estado sólido inicialmente previstos não conseguem operar em frequências da ordem de GigaHertz (microondas). Portanto buscou-se os chaveadores de microondas MW4 (4GHz) da empresa Tyco Electronics, porém nem um representante conseguiu realizar o pedido dos mesmos. Dessa forma, partiu-se para testes e experiências sob própria conta e risco e dessa forma chegou à utilização de relés reversores de 6 pólos da empresa Schrack, com chaveamento mecânico.

A seleção da antena ativa ocorre via microcomandos de software, ativando uma lógica combinacional das saídas digitais do comunicador. Essa lógica combinacional é responsável pela ativação do canal correspondente no multiplexador.

Para realizar o projeto do circuito chaveador de antenas, torna-se necessário conhecer a configuração da pinagem-padrão do conector da antena, conforme ilustração abaixo:

Figura 4 – Representação do conector da antena de leitura/gravação



Fonte: BAUMER IDENT

Abaixo segue a pinagem definitiva utilizada no cabo interligando o comunicador (saída de antena RFID) e o chaveador de antenas.

Cabo Baumer antena (blindado):

pino 1 - branco/azul && branco/verde

pino 4 - branco/marrom

pino 5 - laranja

pino 6 - azul/branco && verde/branco

pino 8 - marrom

pino 9 - branco/laranja

2.2.3.5 Interface comunicador RFID ⇔ Sensores

A ligação de sensores conectados às entradas digitais da interface RFID, bem como o acionamento dos LEDs de sinalização pelas saídas digitais foi realizada através de um cabo contendo um conector DB15 (ver Anexo 2), conforme pinagem a seguir:

==> cabo Entradas 8 vias:

pino 01 - azul 0V neutro

pino 02 - branco/preto IN1

pino 03 - branco/ IN3

pino 04 - verde	IN5
pino 09 - branco/verde	IN0
pino 10 - branco/rosa	IN2
pino 11 - marrom	IN4
pino 12 - laranja	+24VDC positivo
pino 05 - cabo ext. vermelho	+24VDC positivo

==> cabo Saídas 6 vias:

pino 06 - marrom	OUT1
pino 07 - azul	OUT3
pino 08 - verde	OUT5
pino 13 - vermelho	OUT0
pino 14 - laranja	OUT2
pino 15 - amarelo	OUT4

Pinagem do sensor indutivo – Pepperl Fuchs:

pino 1 - brown (marrom) +VCC [9 - 60VDC]
pino 3 - blue (azul) 0V
pino 4 - black (preto) NA (NO normally open 0V)
pino 2 - white (branco) NF (NC normally closed +VCC)

2.2.3.6 Interface PC ⇔ WebCam

Para captação das imagens, foi utilizada uma WebCam LG LIC 300, ligada diretamente ao PC dedicado, via interface USB.

2.2.3.7 Interface PC ⇔ Celular C55 Siemens

Para envio de mensagens SMS utilizando um Celular C55 da Siemens, utilizou-se de um cabo serial, original da Siemens, de código S30880-S5601-A802-1.

2.3 SOFTWARE

2.3.1 Funções

O projeto de software tem como objetivo criar as interfaces de sinais, estabelecer a comunicação entre os componentes de hardware, controlar o fluxo e armazenamento de dados e disponibilizar uma ferramenta de interface com o mundo externo. As ferramentas de software foram escolhidas seguindo as mais atuais tendências de mercado, baseando-se essencialmente em protocolos abertos, software livre e objetos distribuídos. O resultado apresenta-se como uma solução independente de plataforma e interface do usuário baseada em WebServices.

2.3.2 Diagrama em blocos

O projeto de software consiste basicamente de 3 módulos, cuja inter-relação pode ser facilmente visualizada no diagrama a seguir:

- Módulo RFID: comunicação serial através de protocolo EUROX entre comunicador RFID e PC dedicado; comandos de formatação/leitura/gravação de dados através de acesso à estrutura de arquivos do data tag; comandos de acesso às entradas/saídas do comunicador RFID; envio de imagens WebCam;
- Módulo Email/SMS: envio dos dados das cobranças (transações) da base de dados para o proprietário do veículo, através de celular e e-mail.
- Módulo Cliente Web: informações on-line (base de dados, WebCam, RFID).

Figura 5 – Representação do Software em diagrama de blocos



Pelo diagrama acima pode-se observar que o nível de equipamento tem a função de disponibilizar métodos de acesso ao comunicador RFID e à WebCam. Adicionalmente, executa ciclicamente uma thread para verificar a presença de sensores de veículos nas entradas digitais do comunicador RFID. Na prática, o módulo RFID ainda possui a funcionalidade de gravar os créditos de recarga e enviar comandos manuais EUROX. Os aplicativos no nível de aplicação, podem rodar de forma distribuída, bastando acessar a BD.

2.3.3 Ferramentas utilizadas no projeto

No projeto inicial, efetuou-se um estudo a respeito das ferramentas e tecnologias a serem utilizadas no projeto. Abaixo pode-se acompanhar a evolução entre o planejado e o realizado.

Figura 6 – Tecnologias de Software previstas para o projeto

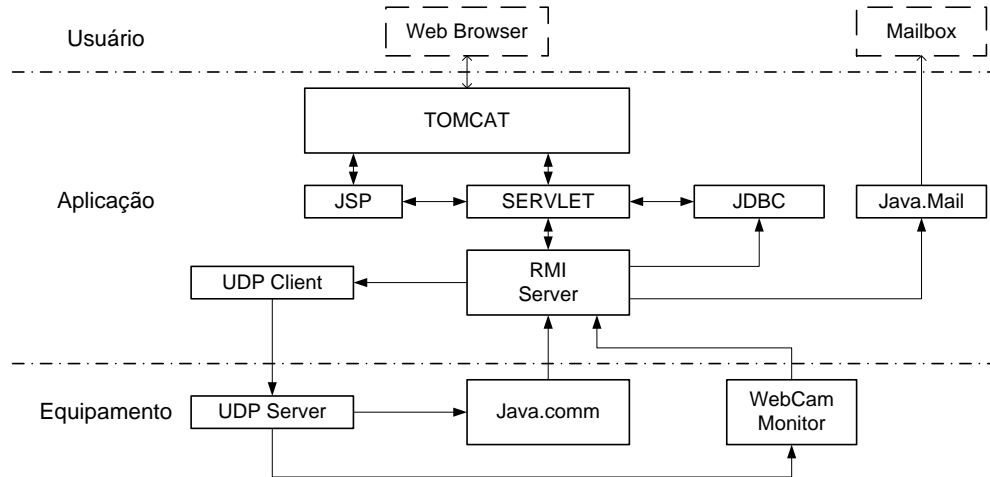
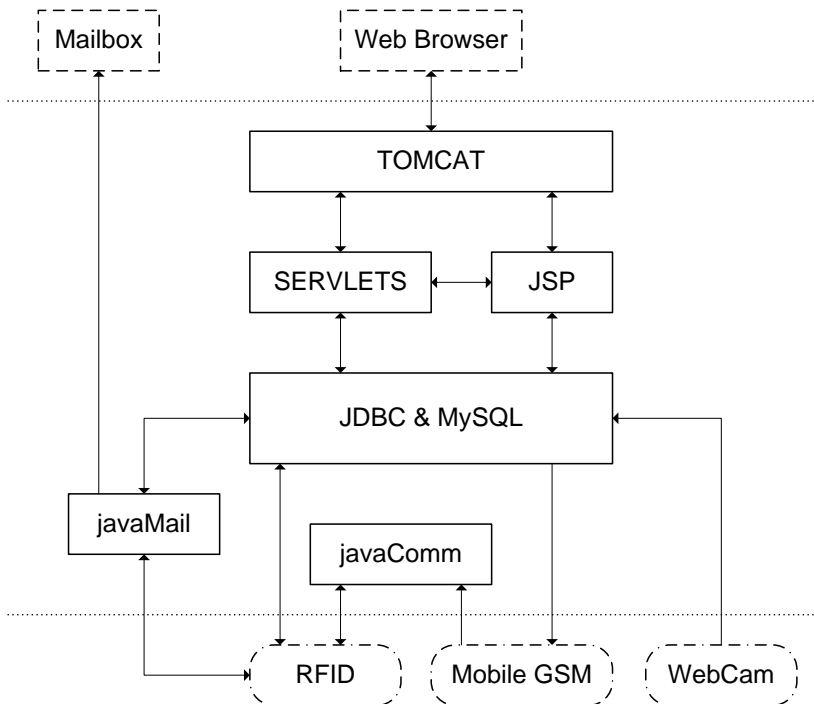


Figura 6a – Tecnologias de Software efetivamente utilizadas no projeto



Para facilitar o entendimento da integração entre os módulos de software, pode-se visualizar na ilustração acima as principais ferramentas utilizadas no projeto, bem como sua inter-relação. Cabe ressaltar que não se encontram listadas ferramentas de baixo nível como micro-códigos ou mesmo o protocolo de comunicação EUROX, utilizado para comunicação entre o PC dedicado e o comunicador de RFID. Em razões de ganho de performance, abandonou-se a utilização de um servidor RMI, passando os aplicativos a acessar diretamente a base de dados através de uma ponte jdbc.

2.4 VALIDAÇÃO DE HARDWARE E SOFTWARE

O processo de validação de Hardware e Software foi feito levando-se em conta a descrição do diagrama tecnológico no item 2.1 - CONCEITOS GERAIS DO PROJETO.

2.5 ESTUDO DE VIABILIDADE TÉCNICO-ECONÔMICA

Realizando-se uma análise funcional do projeto, percebe-se que existe uma coerência na estruturação teórica e na lógica de integração dos módulos de hardware e software. A aplicação de RFID na cobrança de tarifas de veículos automotores em movimento também já existe implementada comercialmente, mesmo que sendo no sistema pós-pago. Portanto o que diferencia a presente especificação dos projetos existentes é o sistema pré-pago, com gravação de créditos no próprio tag, bem como a solução de software através de objetos distribuídos e com interface baseada em WebServices. Um diferencial adicional implementado no projeto foi o sistema de envio de e-mail ao proprietário, com dados e imagem da transação, além do serviço de SMS no celular.

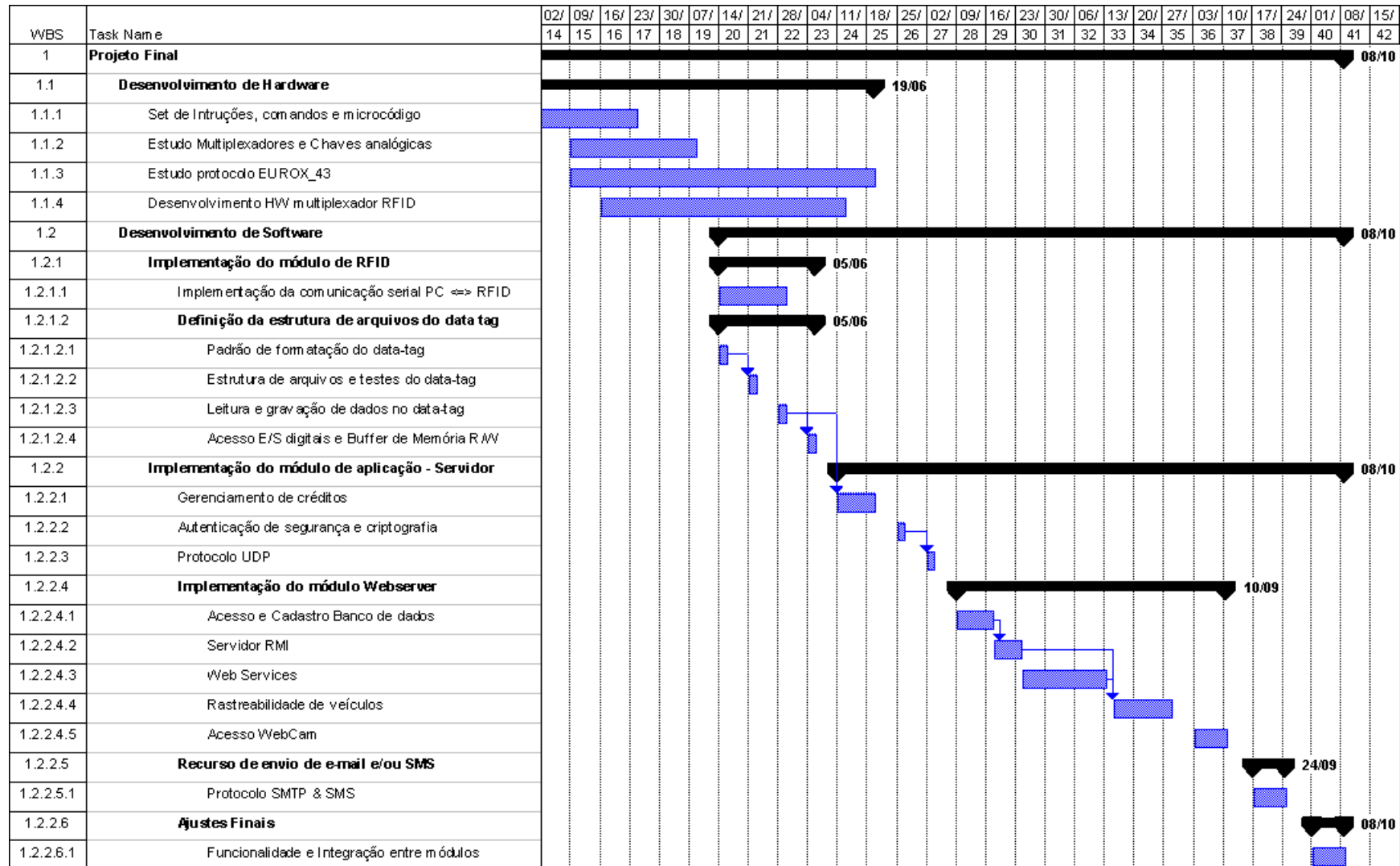
A maior barreira à implementação técnica foi a falta de conhecimentos prévios do autor em interfaces Web, principalmente em se tratando de uma arquitetura baseada em objetos distribuídos e WebServer. Essa dificuldade foi superada através de exaustiva leitura técnica especializada nessa área de conhecimento, bem como dezenas de rotinas de teste escritas até encontrar a solução para os problemas específicos.

Quanto à viabilidade econômica, o projeto se tornou viável graças ao empréstimo de um equipamento de RFID, consistindo de comunicador serial, antena de leitura e escrita, bem como data-tags regraváveis de 32k. O equipamento mencionado, com um custo estimado de R\$19.000,00 representou mais de 95% do custo total do projeto, garantindo a viabilidade econômica da realização do projeto.

Entre os demais materiais de menor valor, cita-se uma WebCam modelo LG LIC-300 USB (R\$ 140,00), um sensor indutivo Pepperl-Fuchs (R\$80,00), um cabo serial Siemens para celular C55 (R\$ 70,00) e componentes eletrônicos diversos (R\$ 150,00).

2.6 CRONOGRAMA

A seqüência de realização de atividades pode ser resumida temporalmente nas tarefas detalhadas no cronograma abaixo:



3 PROJETO DE HARDWARE

3.1 LÓGICA DE FUNCIONAMENTO

Um estudo mais detalhado de hardware possibilita o entendimento da lógica de funcionamento do sistema, através de uma seqüência de procedimentos.

Através de sensores indutivos conectados às entradas digitais da interface de comunicação RFID é possível detectar a passagem de veículos nos locais de controle (cancelas). Através de uma thread de software, verifica-se constantemente o nível lógico destes sensores. Quando o mesmo vai para nível lógico alto, sabe-se em qual local um veículo se encontra, e portanto pode-se selecionar a correspondente antena de leitura. A seleção da antena correspondente à posição do veículo é feita através do envio de um microcomando à interface RFID, contendo o endereço da saída digital a ser ativada.

O relé mecânico conectado à respectiva saída digital estabelece o link físico da unidade central RFID com a antena posicionada sobre o carro. A partir desse momento, ocorre a conexão lógica e o fluxo de dados passa a ser controlado via software.

3.2 ENTRADAS E SAÍDAS DIGITAIS

Aproveitando os recursos existentes na unidade central de RFID, dispõe-se de seis entradas e saídas digitais, operando em níveis de tensão 24VDC.

Fisicamente a conexão dessas E/S pode ser realizada diretamente no comunicador RFID, através de um conector DB15, conforme ANEXO 02.

Quadro 01 – Descrição de Entradas e Saídas Digitais da unidade central RFID

ENTRADAS DIGITAIS	COMPONENTE / FUNÇÃO	SAÍDAS DIGITAIS	COMPONENTE / FUNÇÃO
IN 0	OBJECT DETECT	OUT 0	ALARME ERRO
IN 1	RESET ALARME	OUT 1	OCUPADO R/W
IN 2	INTERLOCK	OUT 2	PC PROCESSANDO
IN 3	SENSOR1 ANTENA READ	OUT 3	LED VERDE
IN 4	SENSOR2 ANTENA WRITE	OUT 4	LED VERMELHO
IN 5	SENSOR WEBCAM	OUT 5	CHAVEADOR

3.3 NÍVEL LÓGICO DE ENTRADAS E SAÍDAS DIGITAIS

Como as entradas e saídas estão conectadas à interface RFID, os níveis lógicos são lidos e enviados através de microcomandos EUROX V4 (encapsulados em Strings Java), e após serem lidos em formato hexadecimal, devem ser convertidos para níveis booleanos.

Descrição	Caractere	Valor ASCII (Hex)
Nível baixo/desligado/inativo	' 0 '	30
Nível alto/ligado/ativo	' 1 '	31
Não alterar nível de saída	' - '	2D

Quando é enviado um comando para as saídas digitais da unidade RFID, o equipamento responde com um *Acknowledge* (ACK), confirmando que o comando foi recebido e sua sintaxe está correta. Do contrário, quando a unidade central enviar um *cancel* (CAN) como resposta, o comando não apresenta uma sintaxe correta ou existe um conflito, como por exemplo acesso a saídas reservadas para o equipamento.

O tempo de estabilização das entradas digitais é de 40ms. Já as saídas são setadas dentro de 5ms após o disparo.

Para realizar a multiplexação das antenas, utilizou-se relés reversores de 6 pólos da empresa Schrack, com chaveamento mecânico, para alta frequência. Conforme mencionado anteriormente, mesmo após exaustivas buscas pelos chaveadores de microondas MW4 (4GHz) da empresa Tyco Electronics, nem um representante conseguiu realizar o pedido dos mesmos.

Com base nesta constatação, e para não alterar o escopo do projeto, decidiu-se utilizar relés mecânicos acionando diretamente os terminais de contato das antenas.

Para evitar ruídos, transientes e harmônicas no chaveamento entre as antenas, decidiu-se primeiro desconectar fisicamente cada um dos contatos de uma antena, para então conectar a segunda antena. Esse procedimento de chaveamento ocorre com a unidade central RFID em funcionamento, porém cabe ressaltar que em hipótese alguma o chaveamento poderá ser efetuado durante uma operação de leitura/escrita de qualquer uma das antenas. A não observância desse procedimento poderá trazer danos irreparáveis ao equipamento, bem como conseqüências inesperadas para algum operador que se encontre nas imediações do equipamento.

Para assegurar a condição acima, foi programado um intertravamento entre as antenas: enquanto existir o valor lógico da saída OUT 1 (*busy*) não é possível realizar a lógica de chaveamento, garantindo que nenhuma antena esteja em processo de comunicação ativa no momento da manobra.

3.4 MATERIAIS UTILIZADOS

Quadro 02 – Descrição Materiais e Equipamentos utilizados no Projeto

Qtde.	Descrição do Equipamento / Fabricante / Modelo
1	Unidade Central RFID (Interface RFID ou Comunicador RFID) Fabricante Baumer Ident; Modelo OIS-P PC3141/03
2	Antena de leitura RFID Fabricante Baumer Ident; Modelo PC3104/32A
3	Cabo de Antena RFID Fabricante Baumer Ident; Modelo PC3117/11A
1	Relé subminiatura com lâminas simples, 6 contatos reversíveis e bobina 24VDC Fabricante Schrack, compatível com Metaltex, Código SBMS6RC3/3A-CIC
1	WebCam USB Fabricante Logitech, Modelo LIC-300
2	Sensores de passagem por lâmina de contato elétrico Fabricação própria
1	Sensor indutivo Fabricante: Pepperl+Fuchs; Modelo VARIKONT
1	Fonte de alimentação 220VAC - 24VDC Fabricante Eurogi; Modelo EAGML00224
1	Autotransformador 127VAC <=> 220VAC (500W) Fabricante Force Line; Modelo 184
1	PC Desktop Pentium IV Fabricante Intel; Modelo P4 1.6 GHz 256 MB Ram
1	PC Desktop Pentium XP 2.4 Fabricante AMD; Modelo XP2400 1GB RAM
-	Cabos de comunicação, conectores e componentes de menor valor

4 PROJETO DE SOFTWARE

As ferramentas de software de alto nível já foram detalhadas na especificação de projeto. No entanto, para um melhor entendimento do fluxo de informações, torna-se relevante demonstrar de forma resumida como ocorre a programação do hardware RFID, o qual foi integrado no presente projeto.

A leitura/escrita dos data-tags (*transponders*) não ocorre de forma direta, como por exemplo em um console MS-DOS. A empresa Baumer Ident, fabricante do hardware de RFID utilizado no presente projeto, disponibiliza três modos de acesso às informações do tag: acesso direto à endereços absolutos de memória, (leitura mais rápida), utilização de blocos e segmentos de leitura seqüencial (velocidade normal) ou mesmo a formatação física com escolha de uma estrutura de arquivos (acesso lento). Neste último modo, deve-se dividir a área de memória do tag em arquivos de tamanho fixo e determinado, os quais uma vez formatados só podem ser alterados com nova formatação.

Como desvantagem temos:

- a. dificuldade de encontrar uma estrutura ideal de arquivos (quantidade x tamanho x seqüência);
- b. elevada complexidade de configuração e formatação (com risco de danos permanentes ao hardware);
- c. pouca flexibilidade: uma vez definida a estrutura de arquivos, não pode-se alterar mais a quantidade, seqüência ou o tamanho dos arquivos sem novo processo de formatação;
- d. tempo de acesso extremamente elevado e velocidade de leitura e escrita de dados consideravelmente baixa em comparação com o acesso a endereços absolutos;
- e. necessidade de mecanismos de criptografia para encapsulamento dos dados.

Apesar de todas desvantagens mostradas acima, a facilidade de uso proporcionada pela implementação de uma estrutura de arquivos chega a justificar todos os argumentos contrários anteriormente citados. Cabe ressaltar que o uso da estrutura de arquivos não inviabiliza o acesso de baixo nível, para ocasiões onde se necessita de velocidade de acesso, leitura e escrita, como é o caso do presente projeto.

Para facilitar a execução de longas seqüências de microcódigo sem impactar na performance, a unidade central de RFID dispõe de buffers de memória para execução rápida de microcomandos.

4.1 PROTOCOLO EUROX V4 - FORMATO DA MENSAGEM

No protocolo Eurox existem quatro tipos de mensagens: Data, ACK, NAK, e CAN, e as diferenças residem no campo de dados da mensagem.

Todas mensagens são compostas de 5 campos:

<STX><ADDR><Data><CSM><ETX>

<STX> “start of message”: um byte de início, com o valor 02(Hex) ASCII

<ADDR> “central unit address”: um byte para endereçamento do comunicador

<Data A> “transmitted data”: número de bytes representando os dados transmitidos

<CSM> “checksum”: dois bytes representando o checksum da mensagem

<ETX> “end of message”: um byte de finalização, com o valor 03(Hex) ASCII

4.1.1 Resumo de caracteres especiais

STX (02 Hex) Início

ETX (03 Hex) Finalização

ACK (06 Hex) Mensagem de reconhecimento

DLE (10 Hex) Utilizado à frente de caracteres especiais no campo de dados

NAK (15 Hex) Conteúdo do campo de dados em mensagem NAK (not acknowledge)

CAN (18 Hex) Conteúdo do campo de dados em mensagem CAN (cancel)

4.1.2 Exemplo de transmissão PC ==> Comunicador RFID

Mensagem “Hello” em forma de campo de dados <DATA>: 'PU0Hello<ETX>'

Valor Hexa	Cálculo CSM	Explicação
02	-	STX (caractere de inicialização)
30	00110000	campo de endereçamento (ASCII 30h == 0dec)
50	01010000	'P', (primeiro byte de dados)
55	01010101	'U'
30	00110000	'0'
48	01001000	'H'
65	01100101	'e'
6C	01101100	'l'
6C	01101100	'l'
6F	01101111	'o'
10	00010000	DLE (filtrar caractere de controle)
03	00000011	ETX (caractere de finalização)
	——XOR——	'Exclusive Or' checksum 01010100
		O resultado do cálculo de checksum (XOR) é 54h
35		'5' (primeiro byte de checksum)
34		'4' (primeiro byte de checksum)
03		ETX (caractere de finalização da mensagem)

Explicação: colocar (to put, ou <<PU>> em Eurox) a string 'Hello' seguida de um caractere de finalização <ETX> no buffer de dados 0 (ASCII 30h == 0dec)

4.2 DOCUMENTOS

4.2.1 Programação orientada a objetos:

A programação do projeto foi inteiramente desenvolvida na linguagem Java, utilizando o SDK 1.5. Intrinsecamente às características da linguagem java, pode-se deduzir que o modelo de software é orientado a objeto. Para a visualização e cadastro de dados via web, utilizou-se do contêiner Apache Tomcat, para executar os servlets e compilar as páginas JSP. Mesmo utilizando-se da plataforma IDE Eclipse 3.02, todos os formulários foram escritos manualmente, a partir da linha de comando. Da mesma forma as páginas para web foram criadas a partir da linha de comando, apenas com comandos html genéricos e portáveis para qualquer navegador (*browser*).

4.2.1.1 Lógica

A lógica de programação segue na íntegra o detalhamento efetuado anteriormente na análise de requisitos.

A cobrança veicular via RFID consiste no processo de leitura do tag, recálculo de créditos e escrita do tag, além do envio de imagem e dados via e-mail e sms.

O módulo de envio de e-mail consiste de uma *thread* que varre ciclicamente a base de dados até encontrar dados de cobrança que ainda não foram enviados ao proprietário. Após confirmação de envio de e-mail, o programa marca o flag de cobrança com status de enviado.

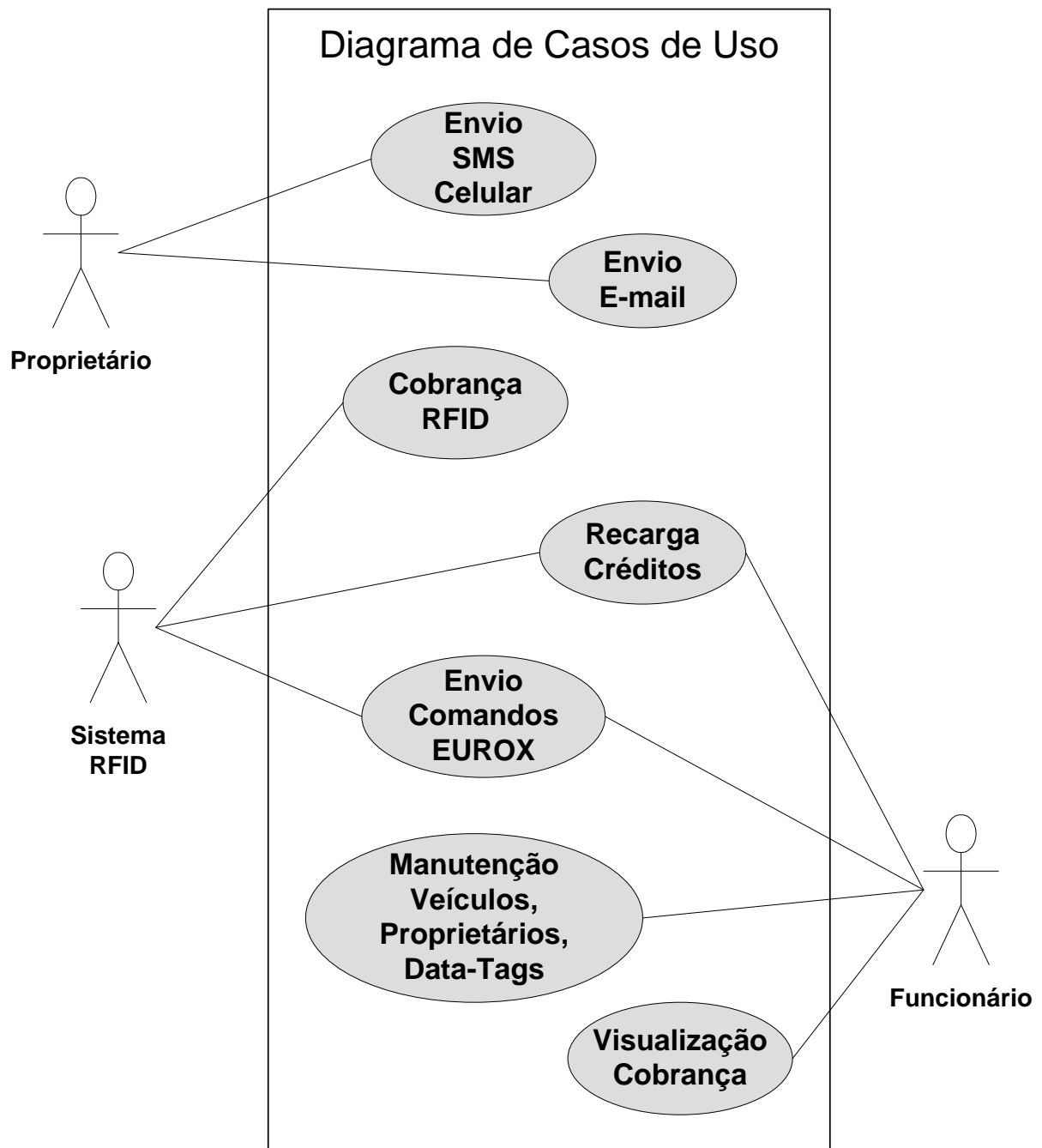
O módulo de envio de sms verifica ciclicamente o status de créditos dos proprietários e a respectiva categoria. Assim que algum cliente estiver com créditos para apenas mais uma transação, envia-se uma mensagem de sms ao celular do proprietário, avisando que os créditos estão acabando, e que na próxima transação aconselha-se a recarga dos mesmos, para evitar um bloqueio por créditos insuficientes.

O procedimento de recarga de créditos envolve a entrada de dados via página web por funcionário qualificado, com posterior gravação dos dados através do mesmo módulo de software utilizado na cobrança RFID.

Da mesma forma, comandos avançados de gravação e escrita, também podem ser enviados manualmente através do programa de cobrança RFID.

Bibliotecas de software não mencionadas na especificação, e que foram utilizadas no projeto: API JavaCOMM para acesso serial ao celular e à interface de comunicação RFID, API JavaMail para envio de *e-mail* com anexos, jconnector para acesso jdbc ao MySQL e biblioteca Morena como *twain driver* para *WebCam*.

4.2.1.2 Diagramas de casos de uso



4.2.1.2.1 Cobrança eletrônica via RFID

CASO DE USO	COBRANÇA ELETRÔNICA RFID
Atores	Sistema de identificação RFID; Funcionário de Estacionamento ou Posto de Pedágio;
Propósito	Visualizar de forma on-line os dados do veículo e os

	créditos disponíveis no data-tag, bem como registrar imagem bitmap do evento de cobrança; Consultar histórico de informações sobre veículos que passaram nos postos de cobrança; Realizar a cobrança eletrônica, com eventual procedimento de recarga de créditos; Repassar um sinal de liberação/bloqueio ao sistema de acionamento e controle da cancela;
Descrição	Realizar a cobrança eletrônica de veículos em movimento; Realizar operações R/W em data-tags RFID, para fins de consulta e atualização de dados; Disponibilizar ao funcionário acesso a dados on-line ou mesmo consulta a históricos de eventos;
Tipo	Primário e essencial
Seqüência de Eventos	
AÇÃO DO ATOR	RESPOSTA DO SISTEMA
<p>1. Este caso de uso inicia quando um veículo ativa o sensor de presença indutivo, nos postos de cobrança automatizada via RFID;</p> <p>2. O sistema RFID aciona a antena de leitura/escrita correspondente ao sensor ativado, bem como registra uma imagem do veículo em movimento, através da webcam;</p> <p>3. Efetua-se a leitura do data-tag;</p> <p>4. Os dados de identificação do veículo são enviados ao sistema;</p> <p>5. Envia-se a quantidade de créditos disponíveis;</p> <p>9. Os créditos são atualizados no data-tag;</p> <p>10. O status final é enviado ao sistema;</p>	<p>6. O sistema verifica e valida os dados vindos dos tags RFID;</p> <p>7. Caso a quantidade de créditos não seja suficiente para efetuar a transação, segue uma solicitação de recarga</p> <p>7.1 A solicitação de recarga de créditos também pode partir voluntariamente do cliente, seguindo então o mesmo fluxo de dados mencionado na ação 7;</p> <p>8. O sistema calcula a quantidade de créditos (débito ou crédito) e repassa ao sistema RFID;</p> <p>11. As informações são persistidas na base de dados.</p>

4.2.1.2.2 Cadastro de Data-Tag's

CASO DE USO	CADASTRAR DATA-TAG's
Atores	Funcionário de Posto de Pedágio ou Estacionamento
Propósito	Registrar e/ou consultar informações sobre Tag's de RFID. Manter um vínculo entre o veículo e o Data

	Tag correspondente
Descrição	Realizar as quatro operações referentes a cadastro de tag's no sistema (inclusão, alteração, consulta e exclusão)
Tipo	Primário e essencial
Seqüência de Eventos	
AÇÃO DO ATOR	RESPOSTA DO SISTEMA
1. Este caso de uso começa quando um funcionário precisa cadastrar DataTag's, dirigindo-se a um terminal Web	2. O sistema solicita a opção de cadastro: inclusão, alteração, consulta e exclusão 3a. Tratando-se de inclusão, solicita ao funcionário entrar com o ID físico do Data Tag a ser inserido 3b. Para alteração e consulta, o sistema solicita a entrada da identificação do Data Tag 3c. A exclusão é prevista apenas para casos excepcionais, como por exemplo uma entrada incorreta de dados, ou o sinistro de um data Tag

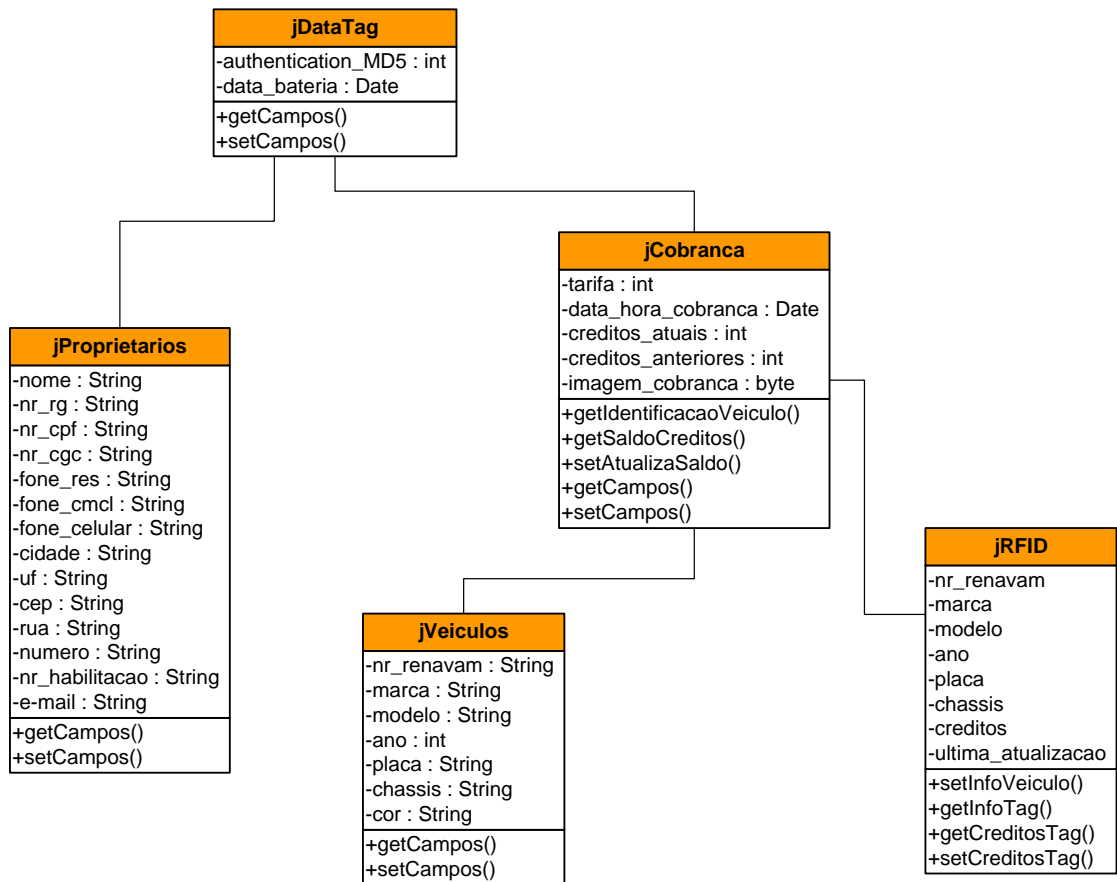
4.2.1.2.3 Cadastro de Veículos

CASO DE USO	CADASTRAR VEÍCULOS
Atores	Funcionário de Posto de Pedágio ou Estacionamento
Propósito	Registrar e/ou consultar informações sobre veículos incluídos no sistema de cobrança RFID. Manter um vínculo entre o veículo e proprietário ou cliente
Descrição	Realizar as quatro operações referentes a cadastro de veículos no sistema (inclusão, alteração, consulta e exclusão)
Tipo	Primário e essencial
Seqüência de Eventos	
AÇÃO DO ATOR	RESPOSTA DO SISTEMA
1. Este caso de uso começa quando um funcionário precisa cadastrar veículos, dirigindo-se a um terminal Web	2. O sistema solicita a opção de cadastro: inclusão, alteração, consulta e exclusão 3a. Tratando-se de inclusão, solicita-se ao funcionário entrar com os dados do veículo (Marca, Modelo, Chassis, Cor, Motor, Proprietário,..) 3b. Para alteração e consulta, o sistema solicita a entrada da identificação do veículo 3c. A exclusão é prevista apenas para casos excepcionais, como por exemplo uma entrada incorreta de dados, ou uma solicitação expressa do cliente

4.2.1.2.4 Cadastro de Clientes/Proprietários

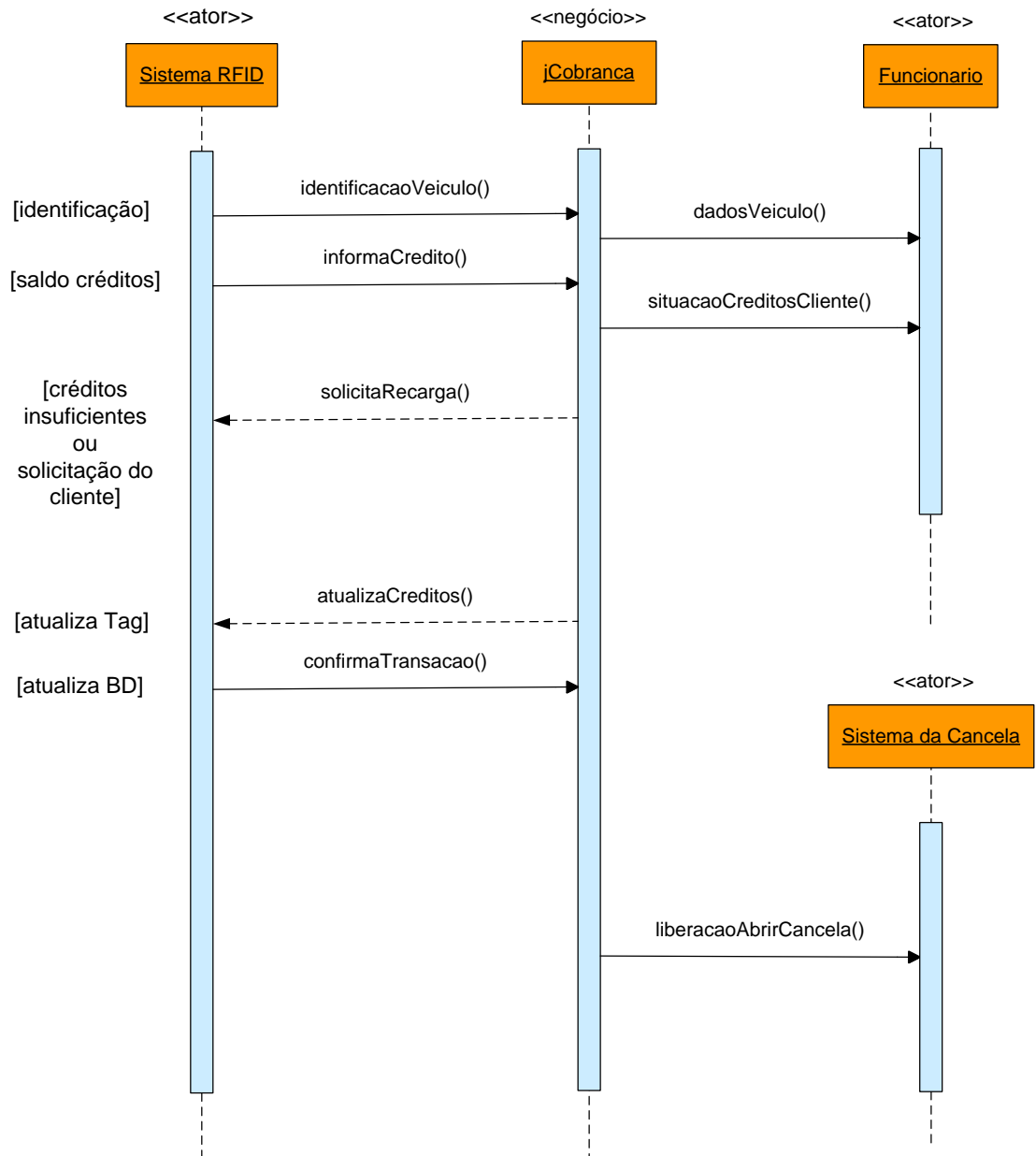
CASO DE USO	CADASTRAR CLIENTES
Atores	Funcionário de Posto de Pedágio ou Estacionamento
Propósito	Registrar e/ou consultar informações sobre clientes/proprietários de veículos que utilizem-se do sistema de cobrança RFID. Manter um vínculo entre o proprietário/cliente e veículos a ele correlacionados
Descrição	Realizar as quatro operações referentes a cadastro de clientes/proprietários no sistema (inclusão, alteração, consulta e exclusão)
Tipo	Primário e essencial
Seqüência de Eventos	
AÇÃO DO ATOR	RESPOSTA DO SISTEMA
1. Este caso de uso começa quando um funcionário precisa cadastrar clientes/proprietários, dirigindo-se a um terminal Web	2. O sistema solicita a opção de cadastro: inclusão, alteração, consulta e exclusão 3a. Tratando-se de inclusão, solicita-se ao funcionário entrar com os dados do cliente/proprietário pessoa física ou jurídica 3b. Para alteração o sistema solicita a entrada da identificação do cliente/proprietário e disponibiliza os campos permitidos para alteração 3c. Quando o funcionário solicita a consulta de dados do proprietário, o sistema disponibiliza também informações referentes às transações eletrônicas ocorridas com os veículos de propriedade ou responsabilidade do cliente 3d. A exclusão é prevista apenas para casos de exceção, como por exemplo uma entrada incorreta de dados, ou um processo de inclusão pendente que não foi aprovado

4.2.1.3 Diagrama de classes

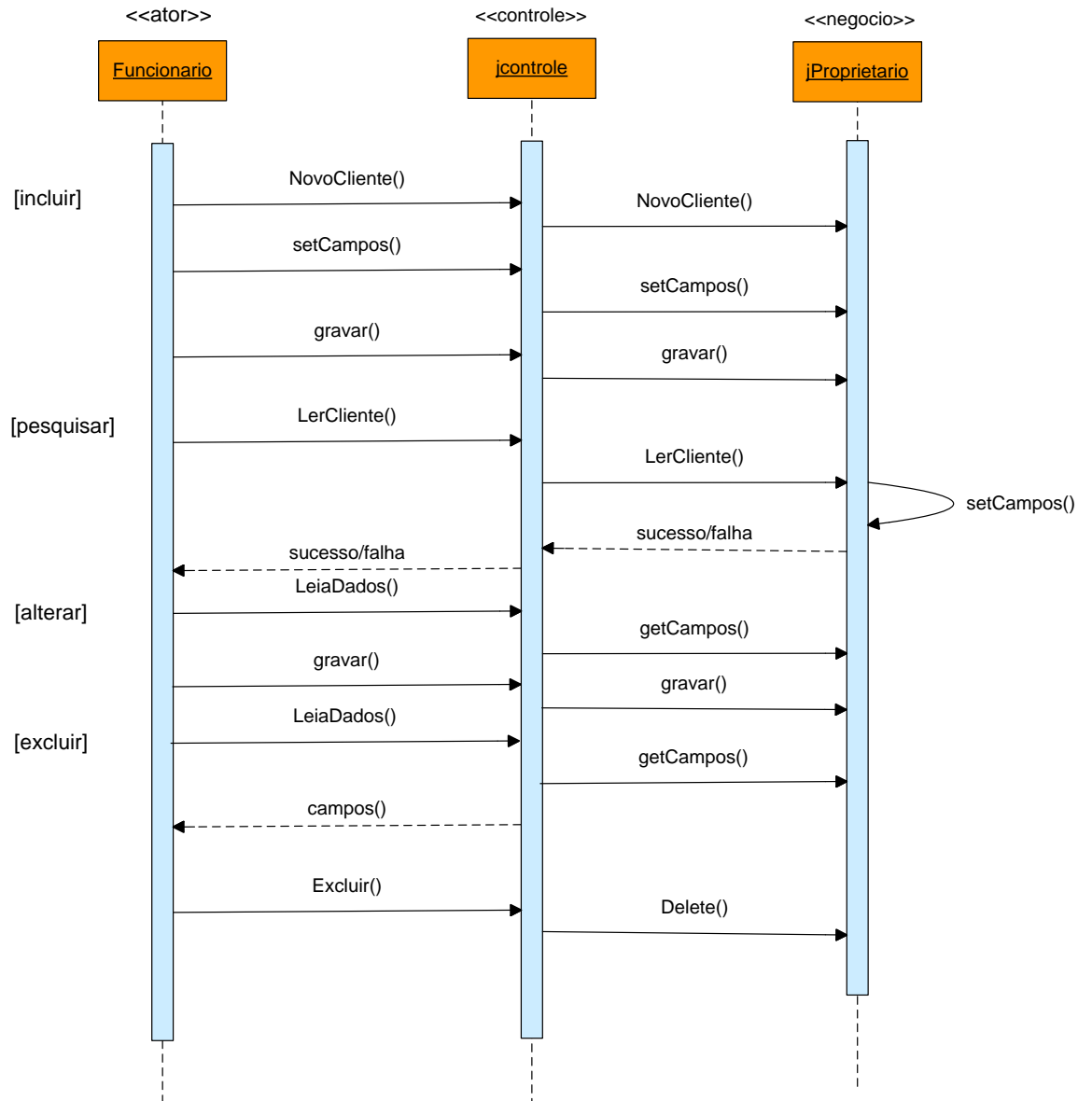


4.2.1.4 Diagramas de seqüência

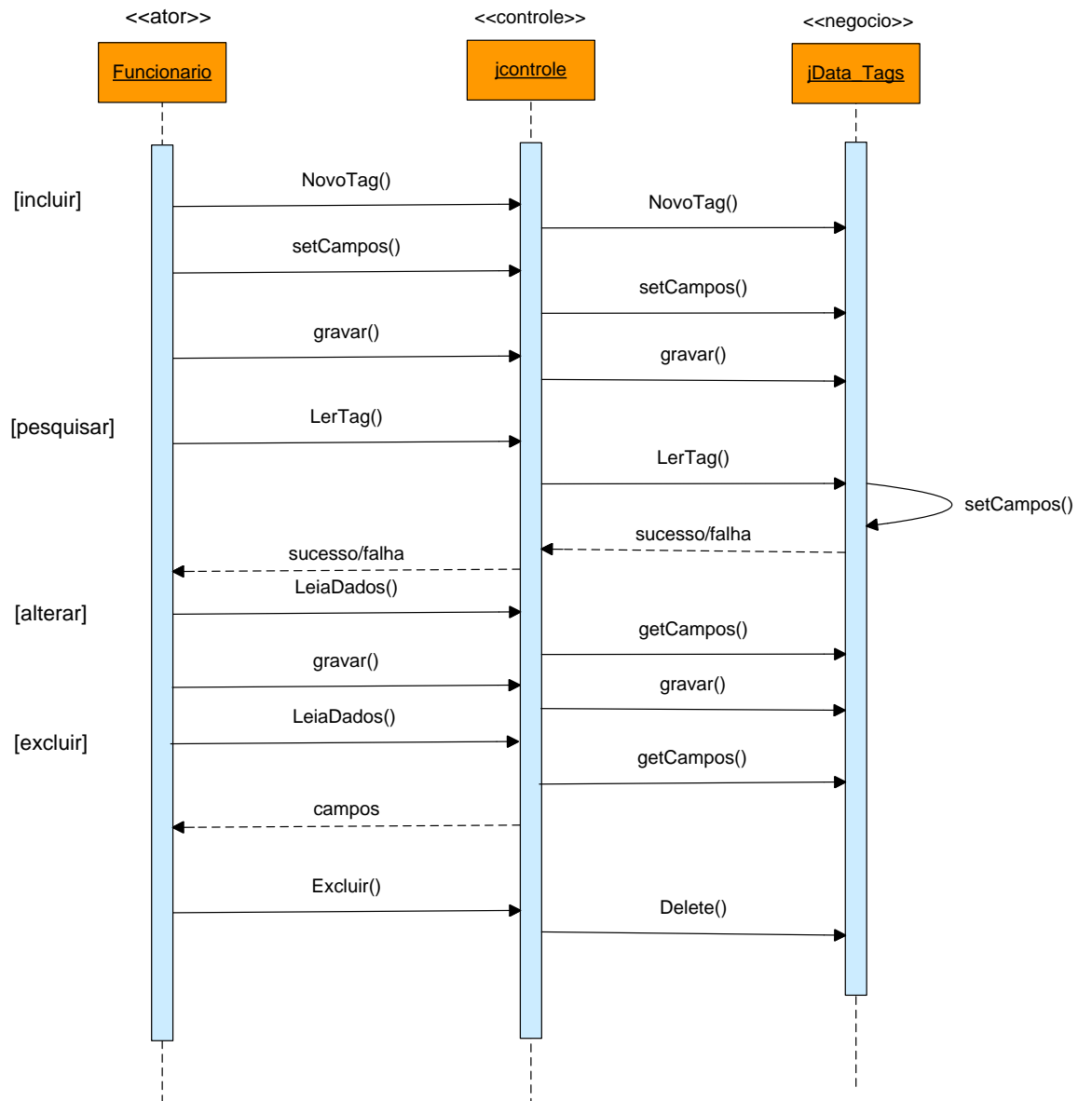
4.2.1.4.1 Cobrança



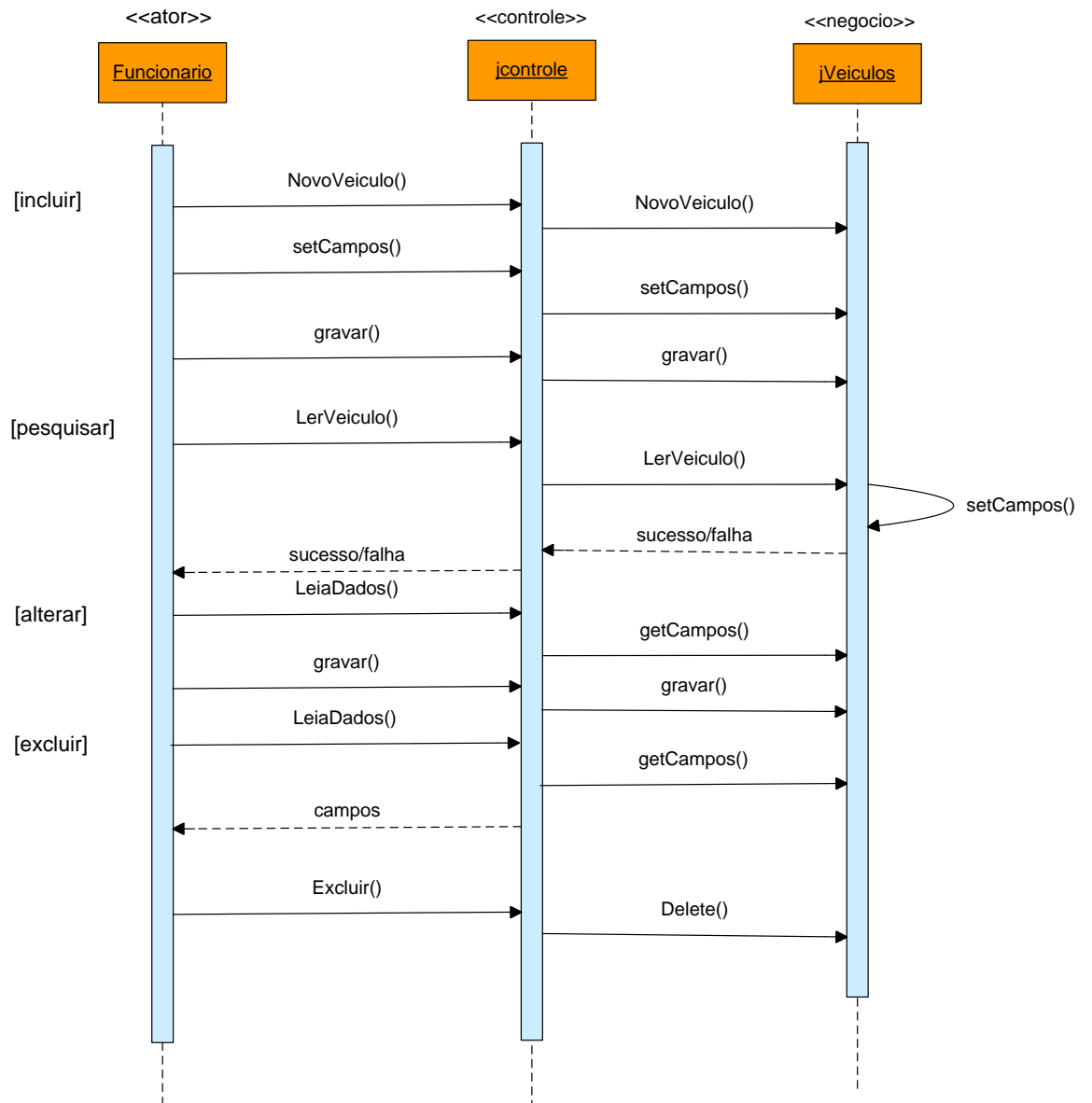
4.2.1.4.2 Cadastro Proprietário



4.2.1.4.3 Cadastro Data-Tags



4.2.1.4.4 Cadastro de Veículos



4.2.1.5 Dados

O sistema gerencial de banco de dados utilizado no projeto foi o MySQL, em razão da popularidade, facilidade de uso e por ser uma ferramenta que evoluiu bastante desde os primórdios do projeto.

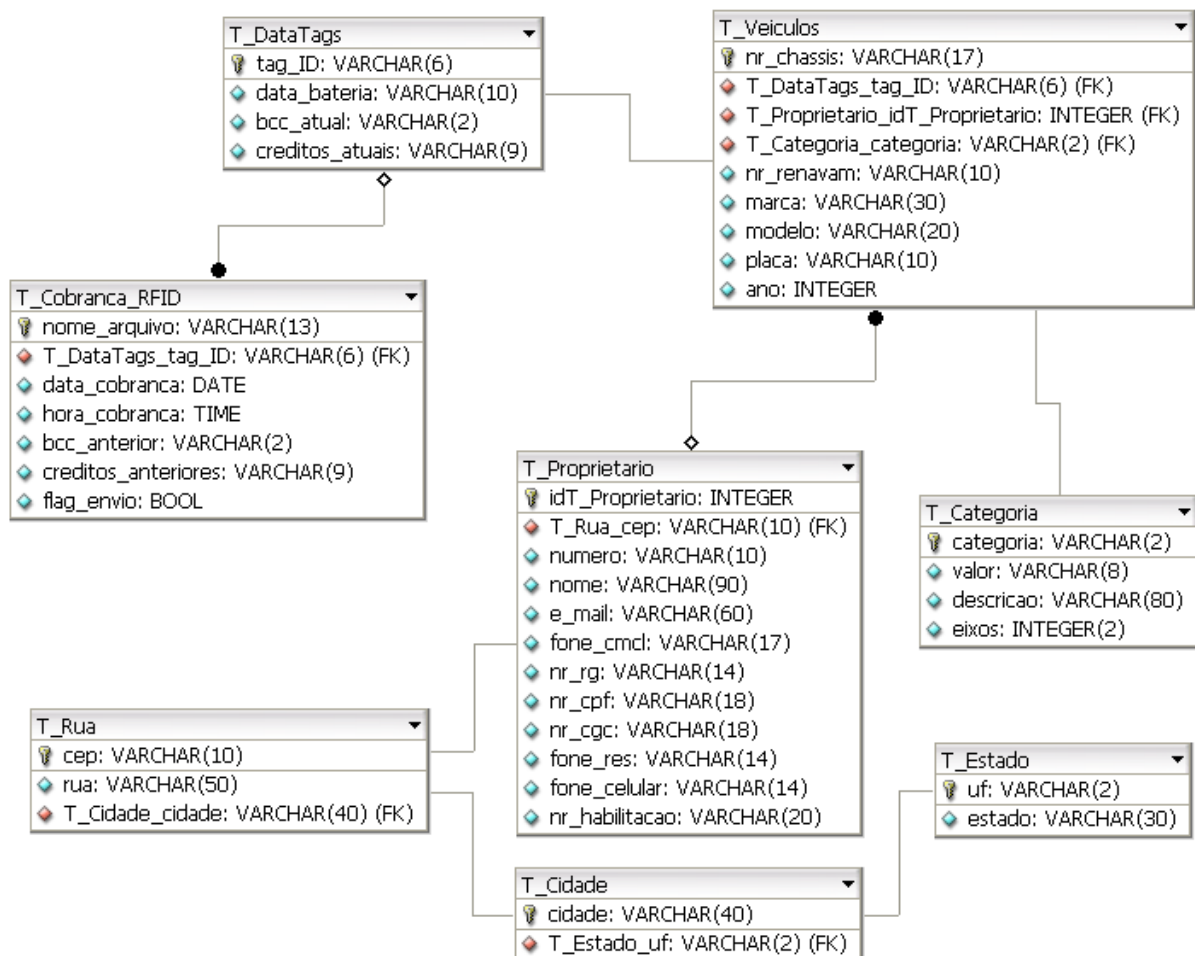
A performance apresentada com 15 acessos simultâneos se mostrou satisfatória.

4.2.1.5.1 Modelo Relacional

“O modelo entidade-relacionamento é baseado numa percepção de um mundo real que consiste em uma coleção de objetos básicos chamados entidades, e em relacionamentos entre esses objetos. Uma entidade é um objeto que é distinguível de outro objeto por um conjunto específico de atributos. Um relacionamento é uma associação entre várias entidades.

Em acréscimo a entidades e relacionamentos, o modelo entidade-relacionamento define certas restrições com as quais os conteúdos de bancos de dados precisam estar de acordo. Uma restrição importante é o mapeamento de cardinalidade que expressa o número de entidades ao qual outra entidade pode estar associada.” [BDGS 99]

Figura 7 – Modelo Entidade-Relacionamento



4.2.1.5.2 Dicionário de Dados

O dicionário de dados é um instrumento para documentar e administrar dados da organização, incluindo identificação sobre características, relacionamentos e autorizações. Em alguns casos, constitui uma ferramenta capaz de armazenar não só metadados (informações sobre os dados), mas também formatos de telas, descrições de relatórios, estruturas de diálogo, associações entre os dados, verificações de validade, controle de segurança, autorizações para leitura ou modificações de dados, campos utilizados para criar campos derivados, faixas permissíveis, relações lógicas entre valores dos dados, etc.

Portanto, os sistemas de dicionários de dados são muito mais do que simples ferramentas de documentação de dados, são complexos sistemas de informação, gerenciamento e controle de dados.

É muito comum associar o dicionário de dados a uma ferramenta de descrição e documentação de dados para posterior utilização por um SGBD (sistema gerenciador de banco de dados). Isto porque o termo dicionário não reflete literalmente toda a sua capacidade e potencialidade.

Tabela nº 04 - T_DataTags

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
tag_ID	Número serial do Data Tag	varchar(6)	(6)	PK
data_bateria	Data da bateria do tag	varchar(10)	(10)	
bcc_atual	Autenticação: block check code	varchar(2)	(2)	
creditos_atuais	Créditos disponíveis do cliente	varchar(9)	(9)	

Tabela nº. 05 - T_Veiculos

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
nr_chassis	Número do chassis	varchar	(17)	PK
T_DataTags_tag_ID	Chave Estrangeira da tabela	varchar	(6)	Not Null
T_Proprietario_idT_Proprietario	Chave Estrangeira da tabela	int unsigned	(10)	Not Null
T_Categoria_categoria	Chave Estrangeira da tabela	varchar	(2)	Not Null
nr_renavam	Numero Renavam	varchar	(10)	
Marca	Placa do veículo	varchar	(30)	
modelo	Modelo do veículo	varchar	(20)	
placa	Placa veículo	varchar	(10)	
ano	Ano do veículo	int unsigned	(10)	

Tabela nº 06 - T_Estado

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
uf	Unidade Federativa	varchar	(2)	PK
estado	Estado Federativo do Brasil	varchar	(30)	

Tabela nº 07 - T_Cidade

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
cidade	Nome da cidade	varchar	(40)	PK
T_Estado_uf	Chave Estrangeira da tabela	varchar	(2)	Not Null

Tabela nº 08 - T_Rua

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
cep	Código de Endereçamento Postal	varchar	(10)	PK
rua	Nome da rua	varchar	(50)	
T_Cidade_cidade	Chave Estrangeira da tabela	varchar	(40)	Not Null

Tabela nº 09 - T_Proprietario

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
idT_Proprietario	Chave Primária da tabela	inteiro	(10)	PK
T_Rua_cep	Chave Estrangeira da tabela	varchar	(10)	Not Null
numero	Número	varchar	(10)	
nome	Nome do proprietário / razão	varchar	(90)	
e_mail	E-mail para envio das notificações	varchar	(60)	
fone_cmcl	Fone comercial	varchar	(17)	
nr_rg	Número do registro geral (RG)	varchar	(14)	
nr_cpf	Número de CPF pessoa física	varchar	(18)	
nr_cgc	Número de CGC pessoa jurídica	varchar	(18)	
fone_res	Fone residencial	varchar	(14)	
fone_celular	Celular	varchar	(14)	
nr_habilitacao	Habilitação do condutor	varchar	(20)	

Tabela nº 10 - T_Cobranca_RFID

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
nome_arquivo	Chave Primária da tabela	varchar	(13)	PK
T_DataTags_tag_ID	Chave Estrangeira da tabela	varchar	(6)	Not Null
data_cobranca	Data da transação	date		
hora_cobranca	Hora da transação	time		
bcc_anterior	Código de autenticação anterior	varchar	(2)	
creditos_anteriores	Quantidade de créditos anteriores	varchar	(9)	
flag_envio	Flag de status de envio	tinyint	(1)	

Tabela nº 11 - T_Categoria

ATRIBUTO	DESCRIÇÃO	TIPO	TAM.	CONST
categoria	Categoria da cobrança	varchar	(2)	PK
valor	Base de valor para a categoria	varchar	(8)	
descricao	Descrição do campo	varchar	(80)	
eixos	Numero de Eixos do veículo	int unsigned	(2)	

RESULTADOS

Apesar de vários imprevistos ocorridos ao longo da fase de implementação do projeto, os resultados alcançados foram bastante significativos e recompensadores. Além de atingir todos objetivos propostos, ainda houve tempo hábil para realizar um criterioso tratamento de exceções de hardware e software, bem como implementar um módulo adicional não previsto inicialmente, constituído pelo envio de mensagens sms ao celular dos proprietários.

O único aspecto que precisaria ser melhorado significativamente para operar comercialmente é o hardware de aquisição de imagens. Provavelmente a webcam tivesse de ser substituída por uma câmera analógica com placa de aquisição digital, e para tanto o software relativo à captação de imagens teria de ser reescrito.

O tempo de leitura e escrita das antenas de RFID ficou em alguns milésimos de segundo, e o tempo de processamento do PC entre leitura e escrita encontra-se na ordem de 1,4 segundos, para um processador Intel P4 1.6GHz com 256MB de memória RAM.

A distância máxima do cabo de antena garantido pelo fabricante é de 100 metros. Com a utilização do multiplexador de antenas, seria teoricamente possível manter uma distância de 200 metros entre a antena de leitura e escrita. Com essa distância, seria possível fazer a cobrança mesmo em auto-estradas européias, sem limite de velocidade. O único detalhe técnico, é que com uma distância tão elevada, o custo dos corredores ficaria muito dispendioso, e também haveria dificuldade de se controlar a passagem seqüencial de veículos, entre os estágios.

Portanto distanciando-se a antena de leitura e escrita em 50 metros, o corredor ficaria com um comprimento otimizado, e seria possível registrar a cobrança de veículos se deslocando a quase 130km/hora.

CONCLUSÃO

Com base nos requisitos e objetivos estipulados, verifica-se que o projeto atingiu plenamente o desempenho esperado. Todas as funcionalidades foram implementadas e atingiram no mínimo o estágio de funcionamento proposto no início do projeto.

A integração entre os módulos não apresentou falhas de sincronia, ou mesmo demoras de atualização. Apenas alguns instantes após a ocorrência de uma cobrança veicular, o módulo de envio de e-mail já busca os dados atualizados na base e envia as informações relevantes ao proprietário, juntamente com uma imagem visual da transação.

Para evitar problemas e ocorrências inesperadas, fez-se um rigoroso tratamento de falhas e exceções. Tentou-se prever a grande maioria de situações possíveis e quase impossíveis de ocorrerem na vida real. Desde o rompimento de cabos de sensores, até falhas na bateria do *transponder*, bem como situações de adulteração das informações do *data-tag*. Dessa maneira, em um hipotético sistema implementado no mundo real, os operadores do sistema teriam a capacidade de apresentar um tempo de reação satisfatório, sem causar insatisfação ou constrangimentos aos clientes. Todas as exceções tratadas apresentam um diagnóstico instantâneo para o operador na tela, e alguns erros são inclusive visualizados ao cliente final, para que o mesmo possa tomar a direção correta, em caso de imprevistos.

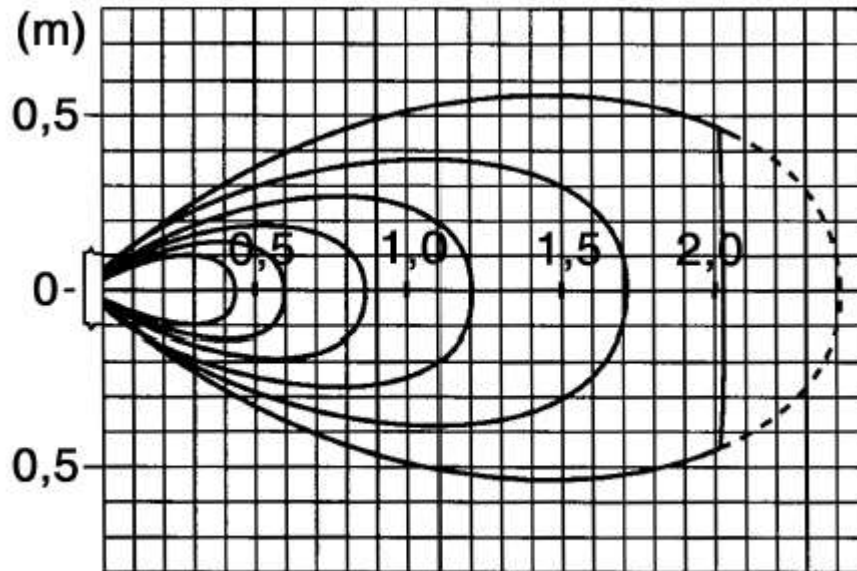
De forma geral, o conceito de software quase permaneceu inalterado, com exceção de algumas ferramentas de *software* que foram simplificadas para otimizar a velocidade de acesso via *web*. Por exemplo o servidor RMI, que previa uma interface de dados em arquivos XML a cada transação, se mostrou ineficiente. Percebeu-se que em razão do elevado volume e frequência de acesso à base de dados, os aplicativos respondiam de forma lenta com a intermediação dos dados em forma de arquivo XML. Portanto optou-se por conexões diretas via JDBC, entre cada aplicativo e a base de dados.

Cabe salientar que uma possível aplicação comercial exigiria o aprimoramento do sistema de captação de imagens. Haveria a necessidade de se utilizar câmeras analógicas de alta velocidade, com placas de captação digital e transmissão via *ethernet*, para suprir a largura de banda necessária para um elevado volume de dados.

ANEXOS

ANEXO 01 – DISTÂNCIA MÁXIMA ENTRE DATA TAG E ANTENA DE LEITURA

Figura 8 – Representação da distância de leitura/escrita da antena 3114/00B

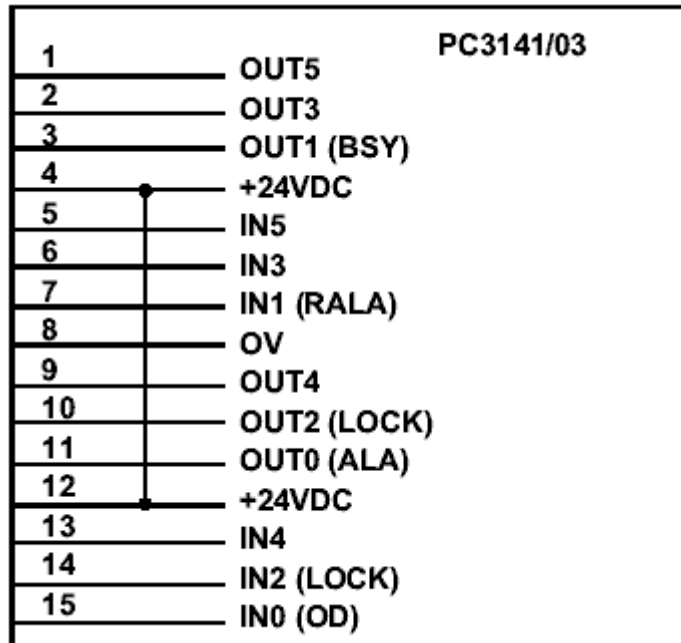


Kommunikationsbereich PC3114/00B

Fonte: BAUMER IDENT

ANEXO 02 – CONECTOR DE ENTRADAS E SAÍDAS DIGITAIS

Figura 9 – Representação do conector de E/S digitais do comunicador RFID



Fonte: BAUMER IDENT

ANEXO 03 – SCRIPT SQL DE CRIAÇÃO DA BASE DE DADOS

Para a criação inicial das tabelas na base de dados utilizou-se do seguinte script SQL:

```
USE rfid;

DROP TABLE if exists T_Veiculos;
DROP TABLE if exists T_Proprietario;
DROP TABLE if exists T_Rua;
DROP TABLE if exists T_Cobranca_RFID;
DROP TABLE if exists T_Cidade;
DROP TABLE if exists T_Categoria;
DROP TABLE if exists T_Estado;
DROP TABLE if exists T_DataTags;

CREATE TABLE T_DataTags (
    tag_ID VARCHAR(6) NOT NULL,
    data_bateria VARCHAR(10) NULL,
    bcc_atual VARCHAR(2) NULL,
    creditos_atuais VARCHAR(9) NULL DEFAULT '100.00',
    PRIMARY KEY(tag_ID)
)
TYPE=InnoDB;

INSERT INTO T_DataTags(tag_ID,data_bateria,creditos_atuais,bcc_atual)
VALUES('150001','2005-08-30','+00100.00','FF');

INSERT INTO T_DataTags(tag_ID,data_bateria,creditos_atuais,bcc_atual)
VALUES('D20120','2000-03-03','+00100.00','FF');

INSERT INTO T_DataTags(tag_ID,data_bateria,creditos_atuais,bcc_atual)
VALUES('C60301','1998-10-11','+00100.00','FF');

INSERT INTO T_DataTags(tag_ID,data_bateria,creditos_atuais,bcc_atual)
VALUES('C603CC','1998-10-08','+00100.00','FF');

CREATE TABLE T_Estado (
    uf VARCHAR(2) NOT NULL,
    estado VARCHAR(30) NULL,
    PRIMARY KEY(uf)
)
TYPE=InnoDB;

INSERT INTO T_Estado(estado,uf)
VALUES('PARANÁ','PR');

INSERT INTO T_Estado(estado,uf)
VALUES('SANTA CATARINA','SC');

INSERT INTO T_Estado(estado,uf)
VALUES('RIO GRANDE DO SUL','RS');

INSERT INTO T_Estado(estado,uf)
VALUES('ESPIRITO SANTO','ES');

INSERT INTO T_Estado(estado,uf)
VALUES('MINAS GERAIS','MG');

INSERT INTO T_Estado(estado,uf)
VALUES('RIO DE JANEIRO','RJ');

INSERT INTO T_Estado(estado,uf)
VALUES('SÃO PAULO','SP');
```

```
UPDATE T_Estado
SET estado = 'SAO PAULO'
WHERE uf LIKE 'SP';

INSERT INTO T_Estado(estado,uf)
VALUES ('DISTRITO FEDERAL','DF');

INSERT INTO T_Estado(estado,uf)
VALUES ('GOIAS','GO');

INSERT INTO T_Estado(estado,uf)
VALUES ('MATO GROSSO','MT');

INSERT INTO T_Estado(estado,uf)
VALUES ('MATO GROSSO DO SUL','MS');

INSERT INTO T_Estado(estado,uf)
VALUES ('ALAGOAS','AL');

INSERT INTO T_Estado(estado,uf)
VALUES ('BAHIA','BA');

INSERT INTO T_Estado(estado,uf)
VALUES ('CEARA','CE');

INSERT INTO T_Estado(estado,uf)
VALUES ('MARANHAO','MA');

INSERT INTO T_Estado(estado,uf)
VALUES ('PARAIBA','PB');

INSERT INTO T_Estado(estado,uf)
VALUES ('PERNAMBUCO','PE');

INSERT INTO T_Estado(estado,uf)
VALUES ('PIAUI','PI');

INSERT INTO T_Estado(estado,uf)
VALUES ('RIO GRANDE DO NORTE','RN');

INSERT INTO T_Estado(estado,uf)
VALUES ('SERGIPE','SE');

INSERT INTO T_Estado(estado,uf)
VALUES ('ACRE','AC');

INSERT INTO T_Estado(estado,uf)
VALUES ('AMAPA','AP');

INSERT INTO T_Estado(estado,uf)
VALUES ('AMAZONAS','AM');

INSERT INTO T_Estado(estado,uf)
VALUES ('PARA','PA');

INSERT INTO T_Estado(estado,uf)
VALUES ('RONDONIA','RO');

INSERT INTO T_Estado(estado,uf)
VALUES ('RORAIMA','RR');

INSERT INTO T_Estado(estado,uf)
VALUES ('TOCANTINS','TO');

CREATE TABLE T_Categoria (
    categoria VARCHAR(2) NOT NULL,
```



```

    valor VARCHAR(8) NULL,
    descricao VARCHAR(80) NULL,
    eixos INTEGER(2) UNSIGNED NULL,
    PRIMARY KEY(categoria)
);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('01','9.80','Automovel Caminhonete Camioneta Furgao',2);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('02','16.40','Caminhao Leve Caminhao-Trator Furgao',2);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('2A','19.60','Onibus 2 Eixos',2);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('03','14.70','Automovel Camioneta ou Caminhonete com semi-reboque',3);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('04','24.60','Caminhao Caminhao-Trator Caminhao-Trator com semi-
reboque',3);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('4A','29.40','Onibus 3 Eixos',3);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('05','19.60','Automovel Camioneta ou Caminhonete com Reboque',4);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('06','32.80','Caminhao e ou Caminhao-Trator com semi-reboque',4);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('07','41.00','Caminhao com Reboque Caminhao-Trator',5);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('08','49.20','Caminhao com Reboque Caminhao-Trator com semi-
reboque',6);

INSERT INTO T_Categoria(categoria,valor,descricao,eixos)
VALUES('09','4.90','Motocicleta Motoneta Bicicletas com motor',2);

CREATE TABLE T_Cidade (
    cidade VARCHAR(40) NOT NULL,
    T_Estado_uf VARCHAR(2) NOT NULL,
    PRIMARY KEY(cidade),
    INDEX T_Cidade_FKIndex1(T_Estado_uf),
    FOREIGN KEY(T_Estado_uf)
        REFERENCES T_Estado(uf)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
)
TYPE=InnoDB;

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Curitiba',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'PR'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Ponta Grossa',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'PR'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Londrina',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'PR'));

INSERT INTO t_cidade(cidade,T_Estado_uf)

```

```

VALUES('Guarapuava',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'PR'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Florianopolis',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'SC'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Joinville',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'SC'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Porto Alegre',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'RS'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Sao Paulo',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'SP'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Campinas',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'SP'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Rio de Janeiro',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'RJ'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Belo Horizonte',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'MG'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Vitoria',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'ES'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Brasilia',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'DF'));

INSERT INTO t_cidade(cidade,T_Estado_uf)
VALUES('Goiania',
(SELECT t_estado.uf FROM t_estado WHERE t_estado.uf LIKE 'GO'));

CREATE TABLE T_Cobranca_RFID (
  nome_arquivo VARCHAR(13) NOT NULL,
  T_DataTags_tag_ID VARCHAR(6) NOT NULL,
  data_cobranca DATE NULL,
  hora_cobranca TIME NULL,
  bcc_anterior VARCHAR(2) NULL,
  creditos_anteriores VARCHAR(9) NULL,
  flag_envio BOOL NULL,
  PRIMARY KEY(nome_arquivo),
  INDEX T_Cobranca_RFID_FKIndex1(T_DataTags_tag_ID),
  FOREIGN KEY(T_DataTags_tag_ID)
    REFERENCES T_DataTags(tag_ID)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
)
TYPE=InnoDB;

CREATE TABLE T_Rua (
  cep VARCHAR(10) NOT NULL,
  rua VARCHAR(50) NULL,
  T_Cidade_cidade VARCHAR(40) NOT NULL,
  PRIMARY KEY(cep),

```

```

INDEX T_Rua_FKIndex1(T_Cidade_cidade),
FOREIGN KEY(T_Cidade_cidade)
REFERENCES T_Cidade(cidade)
ON DELETE NO ACTION
ON UPDATE NO ACTION
)
TYPE=InnoDB;

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Carlos Leinig Junior', '80.820-280',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Floriano', '81.610-000',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Deodoro', '80.010-010',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Floriano', '80.220-000',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Deodoro', '80.060-010',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Floriano', '80.230-110',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Deodoro', '80.050-010',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Floriano', '80.010-130',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Marechal Deodoro', '80.020-320',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

INSERT INTO t_ rua( rua, cep, T_Cidade_cidade)
VALUES('Rua Prof. Pedro Viriato Parigot de Souza', '81.280-330',
(SELECT t_cidade.cidade FROM t_cidade WHERE t_cidade.cidade LIKE 'Curitiba'));

CREATE TABLE T_Proprietario (
idT_Proprietario INTEGER UNSIGNED NOT NULL,
T_Rua_cep VARCHAR(10) NOT NULL,
numero VARCHAR(10) NULL,
nome VARCHAR(90) NOT NULL DEFAULT 'Egon',
e_mail VARCHAR(60) NULL DEFAULT 'egon@milenio.com.br',
fone_cmcl VARCHAR(17) NULL,
nr_rg VARCHAR(14) NULL,
nr_cpf VARCHAR(18) NULL,
nr_cgc VARCHAR(18) NULL,
fone_res VARCHAR(14) NULL,
fone_celular VARCHAR(14) NULL,
nr_habilitacao VARCHAR(20) NULL,
PRIMARY KEY(idT_Proprietario),
INDEX T_Proprietario_FKIndex1(T_Rua_cep),
FOREIGN KEY(T_Rua_cep)
REFERENCES T_Rua(cep)
ON DELETE NO ACTION

```

```

        ON UPDATE NO ACTION
    )
    TYPE=InnoDB;

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(1,'Egon Robert Hubner','04199826625','325','egon@milenio.com.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '80.820-280'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(2,'Adriana C. Thome','++55 41 3317-
3271','0001','adriana.thome@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(3,'Amarildo Reichel','++55 41 3317-
3209','0002','reichel@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(4,'Alessandro Zimer','++55 41 3317-
3270','0003','zimmer@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(5,'Arileide C. Alves','++55 41 3317-
3268','0004','aalves@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(6,'Celso Jose Cordeiro','++55 41 3317-
3271','0005','cjcordeiro@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(7,'Jose Carlos da Cunha','++55 41 3317-
3267','0006','cunha@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(8,'Luiz Carlos P. Albini','++55 41 3317-
3151','0007','albini@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(9,'Marcelo Mikosz G.','04191030936','0008','marcelo@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(10,'Mauricio Perretto','++55 41 3317-
3125','0009','mperretto@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

    INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
    VALUES(11,'Mauricio Schafranski','++55 41 3317-
3151','0010','mschafranski@unicenp.edu.br',
    (SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

```

```

INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
VALUES(12,'Roberto Selow','++55 41 3317-3262','0011','rselow@unicenp.edu.br',
(SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

```

```

INSERT INTO
t_proprietario(idT_Proprietario,nome,fone_cmcl,numero,e_mail,T_Rua_cep)
VALUES(13,'Valfredo Pilla Jr','++55 41 3317-
3270','0012','vpilla@unicenp.edu.br',
(SELECT t_ rua.cep FROM t_ rua WHERE t_ rua.cep LIKE '81.280-330'));

```

```

CREATE TABLE T_Veiculos (
nr_chassis VARCHAR(17) NOT NULL DEFAULT '9BD17103232238676',
T_DataTags_tag_ID VARCHAR(6) NOT NULL,
T_Proprietario_idT_Proprietario INTEGER UNSIGNED NOT NULL,
T_Categoria_categoria VARCHAR(2) NOT NULL,
nr_renavam VARCHAR(10) NULL DEFAULT '792051491',
marca VARCHAR(30) NULL DEFAULT 'BMW',
modelo VARCHAR(20) NULL DEFAULT 'X3',
placa VARCHAR(10) NULL DEFAULT 'MCO8538',
ano INTEGER UNSIGNED NULL DEFAULT '2005',
PRIMARY KEY(nr_chassis),
INDEX T_Veiculos_FKIndex1(T_Proprietario_idT_Proprietario),
INDEX T_Veiculos_FKIndex2(T_DataTags_tag_ID),
INDEX T_Veiculos_FKIndex3(T_Categoria_categoria),
FOREIGN KEY(T_Proprietario_idT_Proprietario)
REFERENCES T_Proprietario(idT_Proprietario)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
FOREIGN KEY(T_DataTags_tag_ID)
REFERENCES T_DataTags(tag_ID)
ON DELETE NO ACTION
ON UPDATE NO ACTION,
FOREIGN KEY(T_Categoria_categoria)
REFERENCES T_Categoria_categoria(categoria)
ON DELETE NO ACTION
ON UPDATE NO ACTION
)
TYPE=InnoDB;

```

```

INSERT INTO t_veiculos(nr_chassis, T_DataTags_tag_ID,
T_Proprietario_idT_Proprietario,
T_Categoria_categoria, nr_renavam, modelo, placa, ano)
VALUES('9BD17103232238676',
(SELECT t_datatags.tag_ID FROM t_datatags WHERE t_datatags.tag_ID LIKE
'150001'),
(SELECT t_proprietario.idt_proprietario FROM t_proprietario WHERE
t_proprietario.e_mail LIKE 'marcelo@unicenp.edu.br'),
(SELECT t_categoria_categoria FROM t_categoria_categoria WHERE t_categoria_categoria
LIKE '04'),
'792051491', 'Furgao Police', 'AR-Y26-937', 2002);

```

```

INSERT INTO t_veiculos(nr_chassis, T_DataTags_tag_ID,
T_Proprietario_idT_Proprietario,
T_Categoria_categoria, nr_renavam, modelo, placa, ano)
VALUES('9BD17103232238677',
(SELECT t_datatags.tag_ID FROM t_datatags WHERE t_datatags.tag_ID LIKE
'D20120'),
(SELECT t_proprietario.idt_proprietario FROM t_proprietario WHERE
t_proprietario.e_mail LIKE 'egon@milenio.com.br'),
(SELECT t_categoria_categoria FROM t_categoria_categoria WHERE t_categoria_categoria
LIKE '01'),
'792051491', 'Cargo', 'KS 54-T', 2001);

```

REFERÊNCIAS BIBLIOGRÁFICAS

- NATIONAL INSTRUMENTS. *What is Serial?* Disponível na Internet: <http://www.ni.com>. Capturado em 15 abr. 2001.
- BAUMER IDENT GmbH. *RF/ID System PC3100/01 8KByte-Kommunikator PC3141/03B Systembeschreibung und Installationshandbuch*. [Descrição do Sistema e Manual de Instalação] Weinheim, Germany, out. 2001. 51p.
- BAUMER IDENT GmbH. *OISP Datenträgersystem PC3100. Systemprogramm EUROX_43 für CPU44*. [Manual de Programação EUROX_43 para CPU44] Weinheim, Germany, set. 1998. 86p.
- [BDGS 99] *Banco de Dados Gerência de Software*. Disponível pela Internet via www: <URL: <http://www.gssof.unicamp.br/gssof/dba/curso-dba.htm>