



Centro Universitário de Brasília – UniCEUB
Faculdade de Tecnologia e Ciências Sociais Aplicadas – FATECS
Curso de Engenharia da Computação

Giuliano Estevam França Boccucci

Sistema de Monitoração e Controle de Acesso
para Condomínios Utilizando a Tecnologia de
Identificação por Rádio Freqüência (RFID)

Brasília
2010

Giuliano Estevam França Boccucci

Sistema de Monitoração e Controle de Acesso
para Condomínios Utilizando a Tecnologia de
Identificação por Rádio Freqüência (RFID)

Trabalho apresentado ao Centro
Universitário de Brasília (UniCEUB) como
pré-requisito para a obtenção de
Certificado de Conclusão do Curso
de Engenharia da Computação

Orientador: Prof. José Julimá Bezerra
Junior

Brasília-DF

2010

Este Trabalho foi julgado adequado para a obtenção do Título de Engenheiro de Computação, e aprovado em sua forma final pela Faculdade de Tecnologia e Ciências Sociais Aplicadas - FATECS.

Prof. Abiezer Amarília Fernandez
Coordenador do Curso

Banca Examinadora:

Prof. José Julimá Bezerra Junior - Mestrado em Engenharia Elétrica - Instituto Militar de Engenharia - Rio de Janeiro - RJ
Orientador

Prof. Vera Lúcia Farini Alves Duarte - Mestrado Matemática - Universidade de Brasília - Brasília - DF

Prof. Thiago de Miranda Leão Toribio - Mestra em Física Teórica - Universidade de Brasília - Brasília - DF

Prof. Flavio Antonio Klein – Mestrado em Estatística e Métodos Quantitativos - Teórica - Universidade de Brasília - Brasília - DF

AGRADECIMENTOS

Primeiramente, a Deus por ter me dado o dom da vida.

Aos meus pais por sempre acreditarem em mim e serem meus principais incentivadores.

A minha esposa por sempre estar do meu lado, me ajudando e me aconselhando.

Aos meus irmãos por confiarem no meu potencial.

Aos meus amigos da empresa Ad-infinitum Soluções, em especial Luis Henrique Barini e Regina Célia Peres Borges, pelo incentivo efetivo na concretização desse projeto, me auxiliando quanto a idéias e espaço físico para o desenvolvimento deste.

Ao Coordenador Abiezer Amarilia Fernandes por estar sempre presente durante a minha formação acadêmica.

Ao meu orientador Professor MC José Julimá Bezerra Junior, pelo apoio irrestrito me auxiliando em todas as etapas do desenvolvimento deste trabalho.

Ao Professor Francisco Javier De Obaldía pelas criticas que possibilitaram a excelência desse projeto.

Aos meus professores que sempre me estimularam com seus exemplos a me tornar um competente engenheiro.

Aos meus colegas de faculdade pela convivência, parceria e amizade, em especial a Aluna Michelle Ribeiro Abuchahin.

LISTA DE ABREVIATURAS

Condominium – Sistema de controle e monitoração de veículos

IP – Internet Protocol

HTML – Hypertext Markup Language

HTTP – HyperText Transfer Protocol

RFID - Radio Frequency IDentification

Rj45 - Registered jack

SQL – Structure Query Language

UHF - Ultra High Frequency

USB – Universal Serial Bus

WWW – World Wide Web

LISTA DE QUADROS

Quadro 1: Frequências dos leitores RFID	07
Quadro 2 – Classes de Tags RFID	09

LISTA DE FIGURAS

Figura 1: Visão geral do sistema	02
Figura 2: Visão geral de um sistema RFID	03
Figura 3: Um sistema RFID.....	05
Figura 4: Leitor RFID	06
Figura 5: Diagrama em blocos de uma etiqueta eletrônica	08
Figura 6: Componentes de uma etiqueta RFID	08
Figura 7: Etiquetas RFID	09
Figura 8: Sistema típico utilizando tecnologia RFID	11
Figura 9: Troca de informações entre antena e etiqueta RFID	12
Figura 10: Componentes Básicos do Sistema <i>Condominium</i>	16
Figura 11: Organograma do Sistema <i>Condominium</i>	17
Figura 12: Netbook Acer – Modelo AOD250-1908	18
Figura 13: Leitora UHF Astra	19
Figura 14: Webcam Creative Nx pro	20
Figura 15: TCOMPORT Corversor Multipoint.....	20
Figura 16: Placa de 8 relês Multipoint	21
Figura 17: Fluxograma funcionalidades do Sistema <i>Condominium</i>	22
Figura 18: Tela de Login	27
Figura 19: Tela Usuários	28
Figura 20: Tela Veículos	28
Figura 21: Protótipo	29
Figura 22: Tela de Confirmação de Entrada ou Saída	30
Figura 23: Tela de Confirmação de Entrada ou Saída	31
Figura 24: Relatório de Entrada e Saída de Veículos.....	31

Figura 25 – Funcionamento da cancela	32
Figura 26 – Funcionamento da cancela	32
Figura 27: Relatório de Entrada e Saída de Veículos	33

SUMÁRIO

Capítulo 1. Introdução	01
1.1 Contextualização do problema	01
1.2 Motivação	01
1.3 Objetivo	02
1.4 Visão geral	02
1.5 Estrutura do trabalho	03
Capítulo 2. Referencial Teórico e Tecnológico	04
2.1 Histórico	04
2.2 O Sistema RFID	05
2.3 Funcionamento dos Sistemas RFID	10
2.4 Principais Vantagens da Tecnologia RFID	12
2.5 Principais Desvantagens da Tecnologia RFID	14
2.6 Aplicações Práticas da Tecnologia RFID	15
Capítulo 3. Sistema de controle de acesso de veículos	16
3.1 Topologia	16
3.2 Especificações do Projeto	17
3.3 Hardware	18
3.4 Integração do Sistema	21
3.5 Operação do Sistema	24
3.6 Protótipo	27
Capítulo 4. Conclusão	34
Referências Bibliográficas	36
Anexos	38
I – Características Técnicas da Leitora UHF Astra	38
II - Creative Webcam NX Pro USB	40
III - Netbook Acer AOD250-1908	41
IV - Placa de 8 relês Multipoint	42

Apêndices	45
A – Estrutura de Relacionamentos do Sistema <i>Condominium</i>	45
B - Scripts de criação do modelo de dados	46
C - Classe de comunicação com a Leitora RFID	49
D - Classe de leitura e gravação das Tags RFID	53
E - Classe para executar a classe cancela	57
F - Classe comunicação cancela	58

RESUMO

Este trabalho apresenta a utilização da comunicação sem fio, obtida por meio da tecnologia RFID (*Radio Frequency IDentification* ou Identificação por Frequência Rádio), como forma de identificação de veículos, criando um sistema computacional que controla o acesso de veículos às áreas comuns de um condomínio. A concretização do projeto se deve à necessidade de aprimorar o nível de segurança e monitoração de veículos nos condomínios pela aplicação da tecnologia RFID. Valendo-se da integração de sistemas, o computador é responsável pelo controle e monitoramento dos periféricos usados neste projeto. Além do RFID, uma câmera de vídeo é utilizada para checagem do condutor pelo porteiro. Este compara a imagem do condutor do veículo, dada pela câmera de vídeo, com a foto cadastrada no computador. O porteiro só deve realizar a abertura do portão se o condutor do veículo estiver cadastrado no sistema. O comando de abertura é realizado por meio de software. Um dispositivo eletromecânico é usado para simular a abertura e o fechamento do portão.

Palavras-chave: RFID; Tags passivas; Controle de acesso; Integração de sistemas.

ABSTRACT

This thesis presents the use of wireless communication thru use of RFID technology (Radio Frequency Identification), to identificate vehicles thru a passive Tag, by designing a computational system that controls the access of vehicles to a condominium' common areas. The fulfillment of the project is a must due the necessity to improve the level of security and control of vehicles in the condominium. Using system integration techniques, the computer is made responsible for the control of the peripherals used in this project. Beyond the RFID, a camera of video it is used to the doorkeeper to check the car driver. The vehicle's driver will be displayed by the video camera and compared with the image of the vehicle's owners photograph stored in the computer. The doorkeeper will open the gate only if the driver is registered in the system. The opening command is accomplished by software. An electromechanical device is used to simulate the opening and the closing of the gate.

Keywords: Passive Tag; RFID; Access control; System integration.

1 INTRODUÇÃO

Os mecanismos de autenticações têm por propósito maior a realização de controle de acesso. A demanda por produtos de segurança de qualidade visando proteger a população dos altos níveis de violência demonstra haver uma nítida necessidade por instrumentos de controle de acesso que incorporem tecnologias de ponta capazes de identificar corretamente quais pessoas e veículos tem acesso a uma determinada área ou setor. Assim, optou-se pelo desenvolvimento de um projeto ligado a área da segurança, visando atender umadas carênciasidentificadas no mercado.

O projeto propõe a utilização da comunicação sem fio, obtida por meio da tecnologia RFID (*Radio Frequency IDentification* ou Identificação por Freqüência Rádio), como forma de identificação de veículos, criando um sistema computacional que controla o acesso de veículos as áreas comuns de um condomínio.

1.1 Contextualização do problema

Devido ao grande aumento da população nas grandes cidades, passou-se a exigir um maior controle de acesso de veículos a adentrar nas garagens dos edifícios e condomínios.

O controle de acesso de veículos a esses locais é um dos maiores problemas encontrados no ramo da segurança nos dias de hoje devido à precariedade dos sistemas disponíveis no mercado. Além de serem falhos na capacidade de informar, tampouco oferecem facilidade de uso e praticidade.

1.2Motivação

A concretização do projeto se deve à necessidade de aprimorar o nível de segurança e monitoração de veículos nos condomínios, e das amplas possibilidades de aplicação da tecnologia RFID.

1.3 Objetivo

O projeto tem por objetivo principal desenvolver um sistema que permita controlar o acesso de veículos às garagens de condomínios. O sistema está integrado a um controle de entrada e saída, oferecendo um serviço mais prático e facilitado para o controle de presença de moradores ou funcionários e identificação de pessoas que estiveram na garagem, assim como da extração de um breve relatório de movimentação pelo local.

1.4 Visão Geral

Este trabalho oferece uma aplicação prática da utilização da comunicação sem fio, obtida por meio da tecnologia RFID, como forma de identificação de veículos, no gerenciamento de acesso a um condomínio.

De um modo geral, a Figura 1 ilustra o fluxo de informações do sistema. O computador recebe informações da câmera de vídeo e do RFID. A câmera de vídeo é utilizada para checagem do condutor pelo porteiro, enquanto o sistema RFID se encarrega de enviar informações cadastrais do condutor. O porteiro compara a imagem do condutor do veículo, dada pela câmera de vídeo, com a foto cadastrada no computador, devendo só realizar a abertura do portão se o condutor do veículo estiver cadastrado no sistema. O comando de abertura é realizado por meio de software. Um dispositivo eletromecânico é usado para simular a abertura e o fechamento do portão.

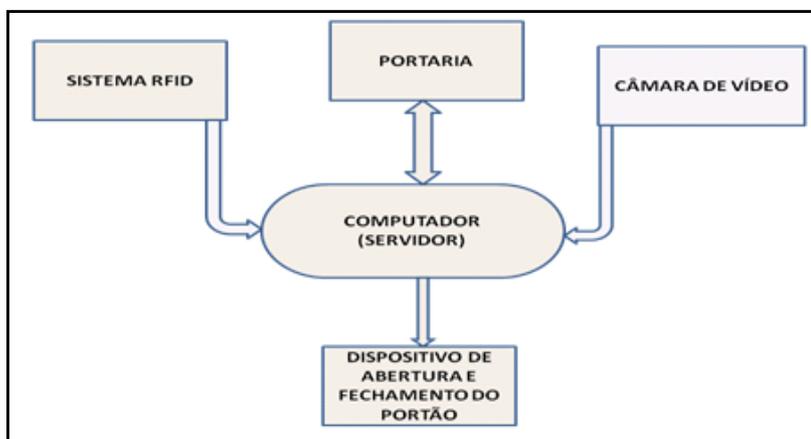


Figura 1: Visão geral do sistema

Na Figura 2 estão identificados os principais componentes de um Sistema RFID: As Tags ou etiquetas, o leitor, a antena e o servidor de aplicação, incluído as formas comuns de conexão entre os componentes.

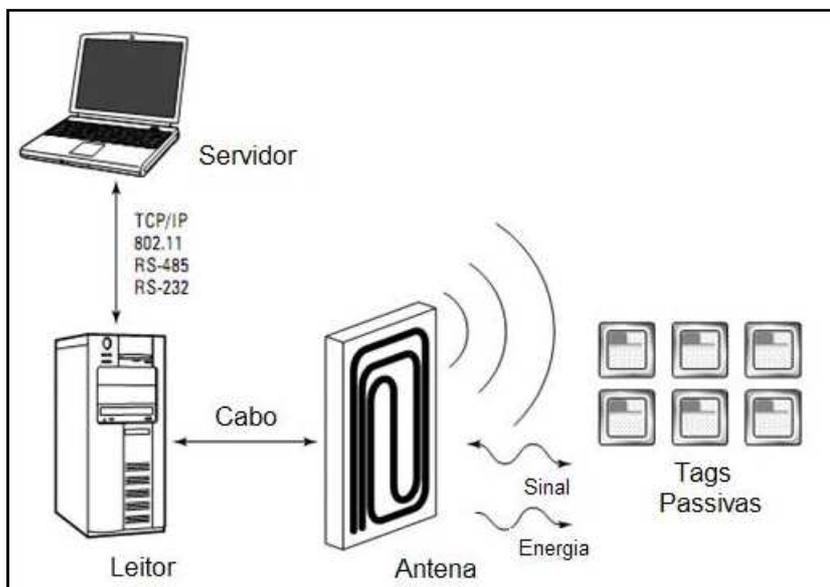


Figura 2: Visão geral de um sistema RFID [9]

1.5 Estrutura do trabalho

Além deste capítulo introdutório, este trabalho tem em sua composição mais 3 capítulos assim descritos:

No capítulo 2 são apresentados o histórico e as informações da tecnologia adotada, o RFID. Informações importantes são passadas sobre os aspectos da tecnologia e sua forma de funcionamento e utilização.

O capítulo 3 aborda o desenvolvimento do projeto, a integração entre o software e o hardware, a base de dados adotada e o software desenvolvido, mostrando todas as informações que são enviadas a leitora.

Por fim, no capítulo 4 são apresentados os resultados obtidos, as dificuldades encontradas e sugestões para trabalhos futuros utilizando a tecnologia RFID.

2 REFERENCIAL TEÓRICO

Neste capítulo é apresentado um breve histórico, formas de leituras e algumas aplicações para a tecnologia RFID. A aplicação da tecnologia está disponibilizada na forma da integração de diversos sistemas físicos já a disposição no mercado e do desenvolvimento de softwares específicos para o melhor funcionamento integrado dos mesmos e da atenção das necessidades de segurança e controle das garagens. O software integrador, a câmara de vídeo e o dispositivo que simula a abertura e fechamento do portão são comentados no Capítulo 3.

2.1 Histórico

Um dos primeiros modelos de identificação por meio de sinais de rádio foi desenvolvido por Sir Robert Alexander Watson-Watt, um físico escocês, no final da década de 1930. Seus desenvolvimentos mostraram-se úteis aos interesses das potências aliadas, assim como as do Eixo, para a detecção prévia de aviões inimigos. Essa tecnologia então emergente é hoje chamada RFID (*Radio Frequency Identification*). [1]

Estes primeiros modelos de uso da tecnologia RFID permitiam distinguir os aviões amigos dos inimigos por meio de artifícios, como, por exemplo, da mudança da orientação dos aviões amigos em determinados pontos do território, modificando, assim, a forma que o sinal retornava aos radares. Esse método relativamente simples é considerado como o primeiro sistema RFID passivo. [1]

Os ingleses, posteriormente na guerra, construíram um sistema mais avançado que incluía um radiotransmissor em seus aviões, que respondia corretamente a um sinal de rádio. Essa nova abordagem para a distinção entre um avião amigo e outro hostil é reconhecida como primeiro sistema RFID ativo. [1]

Já nas décadas posteriores, os interesses acadêmicos e comerciais já não se concentravam na identificação de aeronaves, mas na possibilidade de identificar objetos de modo remoto por meio de ondas de rádio. Essa nova abordagem RFID se baseava no mesmo princípio básico da identificação de aviões, de enviar e receber ondas eletromagnéticas. [2]

Uma das primeiras aplicações práticas do RFID no comércio foi no ramo da segurança, na forma de sistemas antifurto. Por meio de etiquetas colocadas nas mercadorias, era possível determinar se um produto havia sido ou não pago. Essa

forma de vigilância eletrônica continua a ser amplamente utilizada nos comércios. As etiquetas para controle são bastante simples, possuindo uma única informação: pago ou não-pago. Uma etiqueta marcada como não-paga, ao passar pelo raio de ação de um sensor, ativa automaticamente o sistema de alarme para avisar sobre um provável furto.[2] [3]

A tecnologia também tem por característica a atualização constante das etiquetas e do alcance que pode ser devidamente reconhecida e identificada. Os engenheiros da IBM em suas pesquisas durante a década de 1990 haviam ampliado o alcance das etiquetas para 90 metros devido à utilização pioneira de ondas de alta-freqüência (UHF), porém, tais avanços nem sempre se traduziram numa exploração comercial imediata, seja pelo fator “alto custo” ou da falta de interesse do mercado. [4]

O mercado do RFID voltou a se mostrar interessante no final da década de 1990 quando as grandes corporações se decidiram por implementar aplicações logísticas com sistemas RFID com tecnologia UHF para a criação e Centros de Auto-atendimento. As novas etiquetas eram capazes de armazenar uma maior quantidade de informações e serviam plenamente para identificar uma grande variedade de produtos. [4]

2.2 O Sistema RFID

Os componentes da tecnologia RFID são três: Antena, Transceiver (com decodificador) e Transponder (chamado de RF Tag ou apenas Tag), composto de antena e microchip.[8]

A Figura 3 ilustra os componentes básicos de um Sistema RFID.

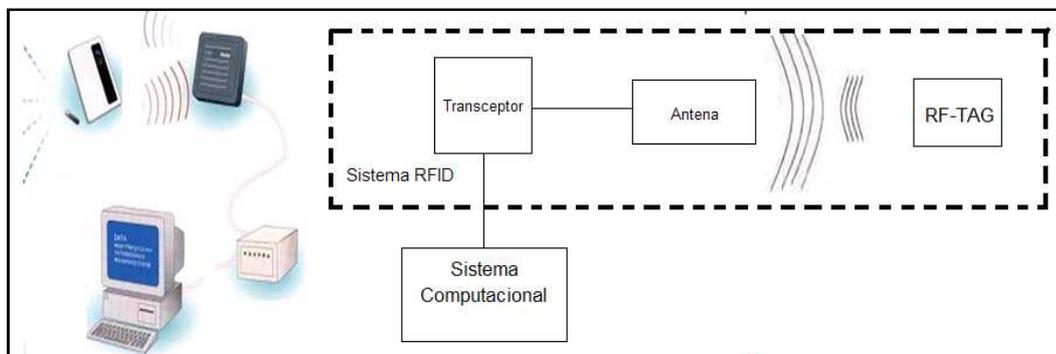


Figura3: Um sistema RFID (Transceptor, Antena, Transponder – RF-TAG e Sistema Computacional) [8]

2.2.1 Antena

A antena ativa o Tag, através de um sinal de rádio, para enviar/trocar informações (no processo de leitura ou escrita). As antenas são fabricadas em diversos tamanhos e formatos, possuindo configurações e características distintas, cada uma para um tipo de aplicação. Quando a antena, o transceiver e o decodificador estão no mesmo invólucro recebem o nome de “leitor”. [5]

2.2.2 Leitores

Este instrumento tem, basicamente, a função de ler etiquetas (Tags), mas modelos mais avançados são igualmente capazes de alterar e processar os dados nelas existentes.

Ao passar pela área de cobertura da antena, o campo magnético da Tag é detectado pelo leitor, decodificando seus dados. Em seguida, o leitor transmite esses dados para um computador para que possam ser processados. [5]

A Figura 4 exemplifica um típico leitor RFID.

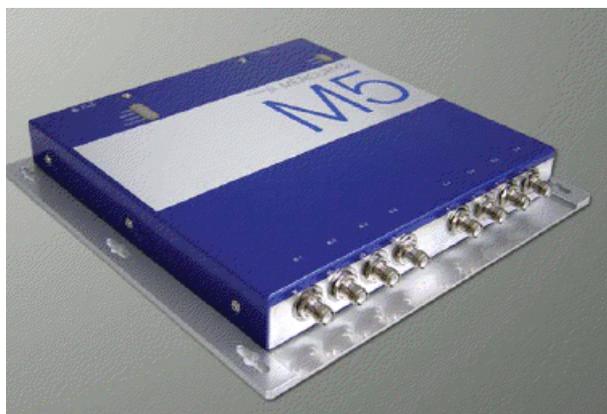


Figura4: Leitor RFID.[19]

O leitor de RFID tem a capacidade de criar um arco eletromagnético capaz de absorver as informações geradas pelas etiquetas para identificar pessoa ou objeto, assim como para converter sinais analógicos em digital. O leitor é dotado de uma antena que capta as informações da etiqueta que passa sobre seu arco e os retransmite para outro dispositivo. No caso de uso conjunto com computadores, as

informações podem ser transmitidas por meio de um cabo rj45, USB, ou por meio de um roteador wireless. [6]

No Quadro 1 são apresentadas algumas das freqüências comumente utilizadas e respectivas aplicações:

Quadro 1: Freqüências dos leitores RFID [7]

FREQÜÊNCIAS	APLICAÇÕES
Baixa freqüência (menos de 135Khz)	<ul style="list-style-type: none">- Identificação de animais- Automação industrial- Controle de acesso
Freqüência Ultra-alta (UHF) (433Mhz e 860 a 930Mhz)	<ul style="list-style-type: none">- Fornece a cadeia de suprimento e logística:- Administração de estoque- Localização de itens- Controle de inventário

2.2.3 Etiquetas RFID (RFID tags)

Nos sistemas de identificação por radiofreqüência, as Tags devem ser consideradas como elemento principal, uma vez que contêm as informações básicas que permitem a identificação física do objeto a ser rastreado. Quando a leitora RFID recebe o sinal da etiqueta, ela retransmite imediatamente os seus dados. [6]

As Etiquetas Inteligentes têm por característica a capacidade de armazenar os dados enviados por transmissores. Elas também respondem a sinais de rádio de um transmissor, transmitindo de volta informações quanto a sua localização e identificação. O microchip envia sinais para as antenas, que capturam os dados e os retransmitem para leitoras especiais. As leitoras selecionam as informações de acordo com o propósito do sistema de Identificação Eletrônica.[5]

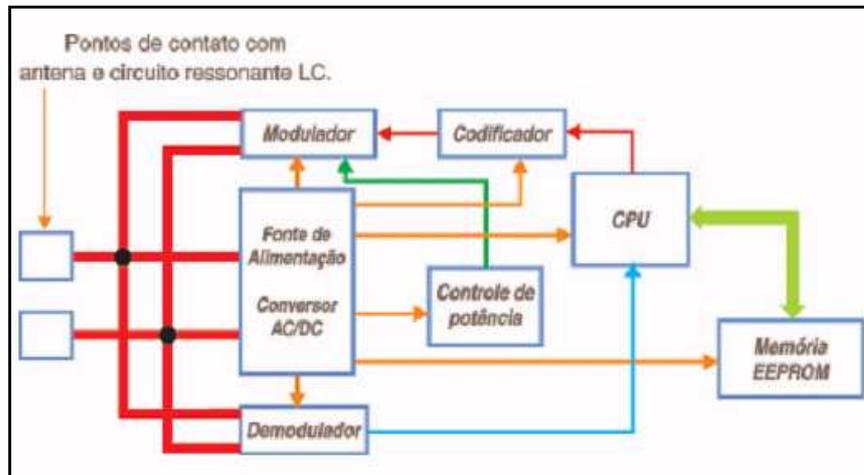


Figura 5: Diagrama em blocos de uma etiqueta eletrônica [6]

Na estrutura da etiqueta representada pela Figura 5 podem-se identificar alguns dos componentes principais: antena ou bobina, microchip, capacitor, diodo e transistor, todos revestidos por uma camada de PVC, resina ou qualquer composto isolante.[7]

A etiqueta pode funcionar tanto como emissora quanto receptora de dados. Sua CPU controla as informações recebidas do Leitor, decodificando-as caso necessário, e seleciona dados e forma pela qual são transmitidos, se codificados ou não, e de acordo com qual sistema de criptografia a sua disposição. [9]

A Figura 6 ilustra com maiores detalhes os componentes antena e microchip de uma etiqueta RFID.

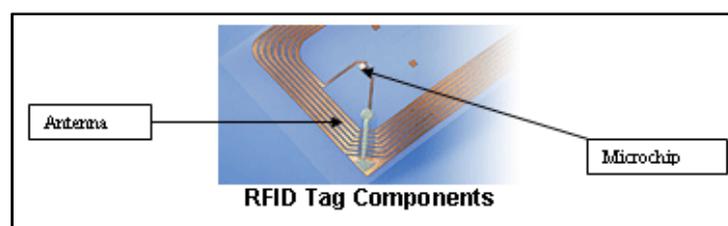


Figura6: Componentes de uma etiqueta RFID [6]

A antena é o componente da etiqueta que reconhece o sinal da leitora. A potência da frequência da onda de rádio fica guardada no diodo que carrega o capacitor, aumentando desta forma a sua tensão.

Quando o capacitor alcança uma determinada tensão pré-determinada, ativa o circuito integrado da etiqueta. É desta maneira que se realiza a leitura do microchip e o seu sinal é enviado para o transistor.

O transistor, por sua vez, gera um campo magnético que será notado pela bobina da leitora RFID e convertido num sinal de baixa frequência. O transistor também pode gerar um campo magnético capaz de ativar a antena para refletir este sinal, o qual será convertido em sinal digital de alta frequência pelo transceptor do leitor [7]

A Figura 7 ilustra as múltiplas formas das etiquetas RFID que estão à disposição no mercado.



Figura7: Etiquetas RFID. [6]

Em relação às funcionalidades da etiqueta RFID, elas são estabelecidas de acordo com uma ordem de classificação[5]

Quadro 2 – Classes de Tags RFID [5]

CLASSE DA ETIQUETA	DESCRIÇÃO E FUNCIONALIDADE
Classe 0/I:	Etiquetas passivas – leitura
Classe II:	Etiquetas passivas com funcionalidades adicionais como memória e criptografia
Classe III:	Etiquetas semi-passivas com suporte e comunicação em banda larga
Classe IV:	Etiquetas ativas com capacidade de comunicação em banda larga ponto a ponto com outras etiquetas ativas que operem na mesma frequência e com leitores
Classe V	Etiquetas desta classe são essencialmente leitores, pois podem ter etiquetas das classes I, II e III, assim como se comunicar com etiquetas da classe IV e qualquer outro dispositivo wireless

Existem, então, três tipos de etiquetas RFID: passivas, ativas e semi-passivas. [6] [7]

2.2.3.1 Etiquetas RFID passivas

Este tipo de etiqueta se caracteriza por não precisar de nenhum tipo de alimentação interna. A energia necessária para o seu funcionamento está localizada no seu próprio sistema de leitura, que a produz por meio de indução magnética ou campo eletromagnético. Quanto ao seu alcance, dificilmente ultrapassa os 5 metros de distância. É o tipo de etiqueta mais utilizado porque seu custo é extremamente acessível. [6] [7]

2.2.3.2 Etiquetas RFID ativas

As etiquetas ativas possuem fonte de energia interna e também um transmissor. Seu alcance pode ser medido na escala de quilômetros. Devido ao seu alto custo de produção são utilizadas em sistemas especiais. [6] [7]

2.2.3.3 Etiquetas RFID semi-passivas

As etiquetas semi-passivas, como as ativas, possuem uma fonte de alimentação interna, mas não têm a capacidade de transmitir dados, apenas de recebê-los. Devido a essa particularidade, podem ser utilizadas em ambientes cujo campo magnético é mais fraco, evitando, também, interferências externas para o seu melhor funcionamento. A sua bateria interna lhes permite ser identificadas a distâncias de até 100 metros. Assim como as etiquetas ativas, o custo mais elevado de produção restringe sua utilização em larga escala. [6] [7]

2.3 Funcionamento dos Sistemas RFID

Os leitores, como dito anteriormente, com o auxílio da antena, que é o seu componente vital, transmitem ondas de radiofrequência que se propagam até o

limite de potência/tensão fornecido pela antena. As tags, por sua vez, absorvem energia necessária para conseguir transmitir suas informações, por meio de uma bobina indutora nela presente. [5] [6] [7]

A Figura 8 apresenta uma visão geral do funcionamento de um Sistema RFID. As Tags podem ser utilizadas tanto no controle de estoque de mercadorias quanto para rastrear o deslocamento físico de uma carga e de um veículo.

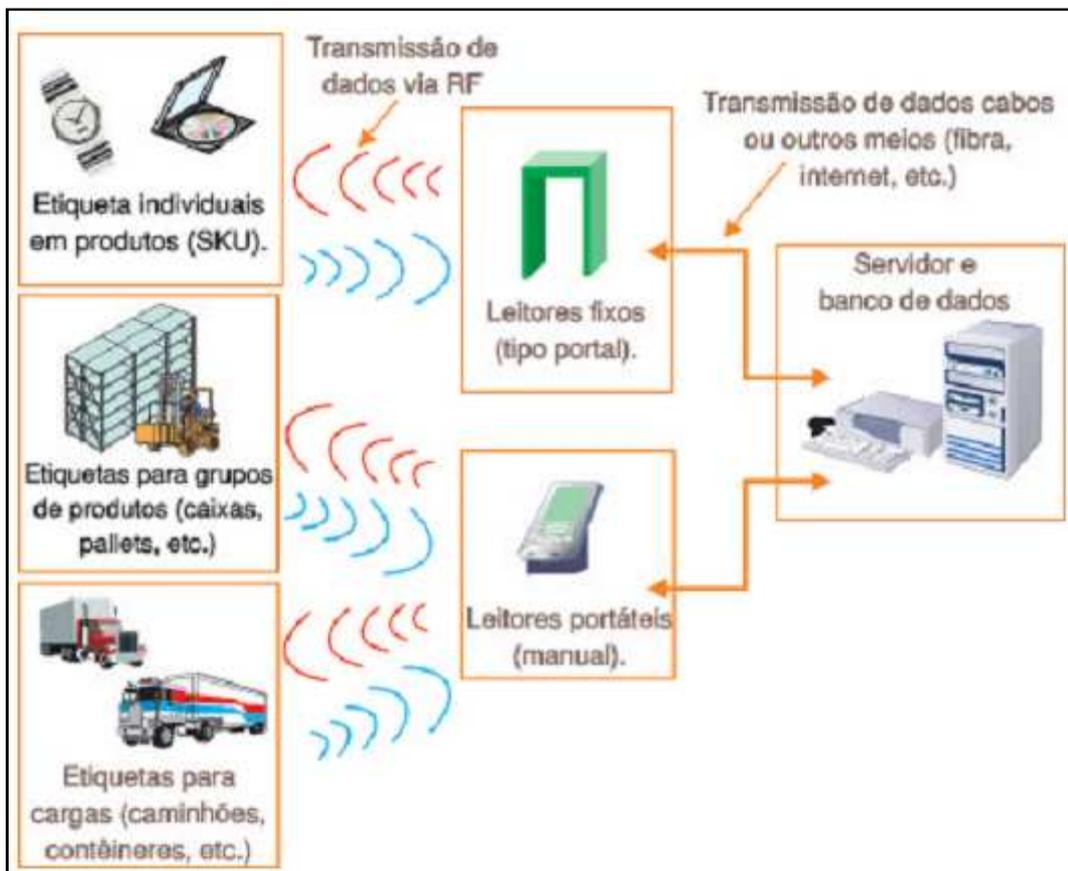


Figura 8: Sistema típico utilizando tecnologia RFID [6]

A energia consegue ser captada e fornecida para o chip que processa e faz o caminho inverso da energia, passando os dados para a leitora. O leitor, por sua vez, encaminha os dados a um computador que está com um sistema pronto para receber as informações contidas nas tags, tais como sistema de gestão, sistema de relacionamento com clientes (RCM), sistemas de suprimentos, sistemas de identificação eletrônica de pessoas, objetos, sistemas de estacionamentos, entre outros. [6] [8]

A Figura 9 esquematiza como ocorre a seqüência de comunicação entre uma Tag, a antena e o leitor RFID. Conforme referido anteriormente, a Tag ao entrar

no alcance da antena/leitor envia automaticamente as informações para sua identificação.

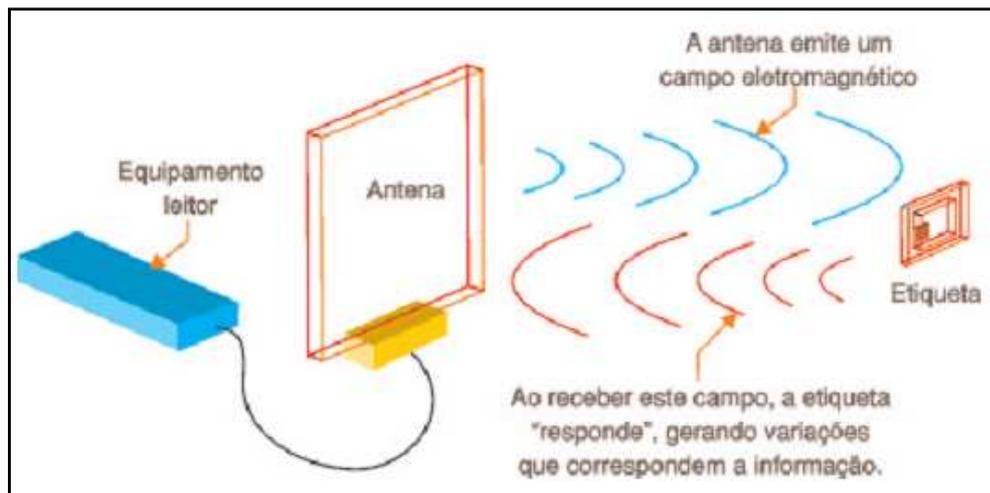


Figura9: Troca de informações entre antena e etiqueta RFID [6]

Esses sistemas, por incluírem bancos de dados em seus projetos, conseguem acessar em tempo real as informações neles anteriormente cadastradas, diminuindo o tempo de processamento e agilizando o tempo de resposta. [5] [6] [8] [9]

2.4 Principais Vantagens da Tecnologia RFID

Entre as principais vantagens da tecnologia RFID estão: [6]

- a) Tecnologia sem fios: há uma comunicação bidirecional entre leitor e etiquetas completamente sem fios. Apenas esta característica já elimina uma série de custos envolvidos em sistemas cabeados. Isto elimina também a necessidade de conexão com a etiqueta para efetuar a leitura e gravação da mesma.
- b) Tecnologia reciclável: as etiquetas podem ser utilizadas nas mais diversas aplicações. Algumas etiquetas permitem a gravação e leitura no ambiente de utilização. Isto significa dizer que uma etiqueta pode ser regravada diversas vezes e reaproveitada em outros objetos após o uso.
- c) Tecnologia robusta: por sua construção mecânica não ter partes móveis e ser lacrada, não permitindo acesso ao chip, as etiquetas podem ser aplicadas nas mais diferentes condições de umidade, calor, ambientes

corrosivos (com tratamento adequado), etc. e ainda assim serem extremamente funcionais.

d) Capacidade de armazenamento de dados: caso os chips possuam memórias, dados referentes ao produto podem ser gravados, contendo desde o histórico do mesmo até as configurações e particularidades de cada um. Quanto maior a memória, mais dados e maior a capacidade de gerenciamento de informações que as etiquetas proporcionam.

A tecnologia RFID proporciona também estas vantagens práticas em aplicações logísticas: [7]

a) Rapidez;

b) precisão e confiança na transmissão de dados;

c) Elevado grau de controle e fiscalização, que aumenta a segurança e evita furtos além de evitar falsificações de mercadorias;

d) Possibilidade de leitura de muitas etiquetas de forma simultânea e captação de ondas à distância;

e) Identificação sem contato nem visão direta do produto, que possibilita a codificação em ambientes hostis;

f) Simplificação dos processos do negócio, que permite a redução da força de mão de obra com transferência dos atuais empregados nestas atividades para atividades mais nobres;

g) Rastreabilidade de produtos (controle de inventário) e de informação (ciclo de vida), que acarretam uma melhoria nas operações de gerenciamento e controle;

h) Alta capacidade de memória, que propicia o armazenamento de todas as informações pertinentes;

i) Leitura e escrita, que criam a possibilidade de constante atualização dos dados recebidos e

j) Durabilidade do estoque com possibilidade de reutilização.

2.5 Principais Desvantagens da Tecnologia RFID

Como visto anteriormente, as tecnologias RFID oferecem uma série de vantagens de produção e de tecnologia. Porém, podem apresentar algumas dificuldades para sua melhor utilização.

Entre as principais desvantagens ao uso da tecnologia estão: [5]

- a) O custo elevado da tecnologia RFID em relação aos sistemas de código de barras é um dos principais obstáculos para o aumento de sua aplicação comercial;
- b) O preço final dos produtos, pois a tecnologia não se limita ao microchip anexado ao produto apenas. Por trás da estrutura estão antenas, leitoras, ferramentas de filtragem das informações e sistemas de comunicação;
- c) O uso em materiais metálicos e condutivos relativos ao alcance de transmissão das antenas. Como a operação é baseada em campos magnéticos, o metal pode interferir negativamente no desempenho;
- d) A padronização das frequências utilizadas para que os produtos possam ser lidos por toda a indústria, de maneira uniforme; e
- e) A invasão da privacidade dos consumidores por causa da monitoração das etiquetas coladas nos produtos. Para esses casos existem técnicas de custo alto que, quando o consumidor sai fisicamente de uma loja, a funcionalidade do RFID é automaticamente bloqueada.

Outras deficiências de aplicação percebidas na tecnologia foram: [6]

- a) Física: ondas de rádio frequência (RF) não se propagam facilmente em qualquer ambiente. Isto limita algumas aplicações dos sistemas.
- b) Ambiente: interferências de outros sistemas, excesso de calor, tempestades solares, eletricidade estática. Tudo isso pode mudar as condições de propagação ou maquiagem os dados trocados entre antenas e etiquetas.
- c) Precisão: colisão de dados lidos, distância entre antena e etiqueta, condições ambientais entre outros são componentes que prejudicam a precisão das operações de leitura e escrita de dados.

d) Segurança: as informações de todas as etiquetas podem ser lidas por qualquer um que tenha um leitor de dados idêntico aquele utilizado pelo sistema. Isso faz necessário criar códigos de segurança para objetos importantes.

e) Normatização: como ainda existem diversos padrões disponíveis no mercado, existe uma possibilidade de adquirir um sistema RFID que seja incompatível com o do seu principal fornecedor.

f) Maturidade: esta tecnologia ainda se encontra em fase de constante desenvolvimento. Novidades podem surgir a cada instante e mudar toda a tecnologia adquirida.

2.6 Aplicações Práticas da Tecnologia RFID

Os sistemas RFID são considerados de grande utilidade para a logística das organizações. Entre as possíveis aplicações estão o controle da origem e qualidade de alimentos, controles alfandegários, traslado de bens e mercadorias, sistemas de segurança anti-roubo e controle de estoques. [8]

Muitas são as áreas onde a tecnologia RFID pode ser aproveitada. Por exemplo: na área de segurança para o monitoramento de cargas e no acesso a áreas restritas, na agropecuária, nas linhas de produção, para o controle de estoque e controle de bagagens em aeroportos. [5]

A tecnologia RFID também oferece soluções para o controle de passaportes, controle de entrada e saída de livros de bibliotecas e controle do transporte de madeiras. Na área da saúde as etiquetas inteligentes podem advertir desde uma galinha específica num galinheiro cuja temperatura corporal está mais elevada do que o normal até o controle do ritmo cardíaco de um indivíduo, alertando imediatamente os serviços hospitalares indicando a sua presente localização. [9]

A tecnologia RFID também encontrou espaço em áreas da produção de veículos, na distribuição de botijões de gás e controle de ponto e controle de patrimônio. [7]

3 SISTEMA DE MONITORAÇÃO E CONTROLE DE ACESSO PARA CONDOMÍNIOS

O Sistema de Monitoração e Controle de Acesso para Condomínios está estruturado numa proposta de utilização da tecnologia RFID. O controle de entrada e saída de veículos além de oferecer uma maior segurança aos condôminos, disponibiliza ao administrador do condomínio alguns relatórios que facilitam a sua logística tanto para o melhor agendamento de pessoal para atender os horários de maior demanda quanto para assegurar os interesses do condomínio quanto a danos e roubo de veículos.

Este sistema foi batizado “Sistema *Condominium*”.

3.1 Topologia

A Figura 10 a seguir disponibiliza uma visão geral dos componentes do Sistema *Condominium*.

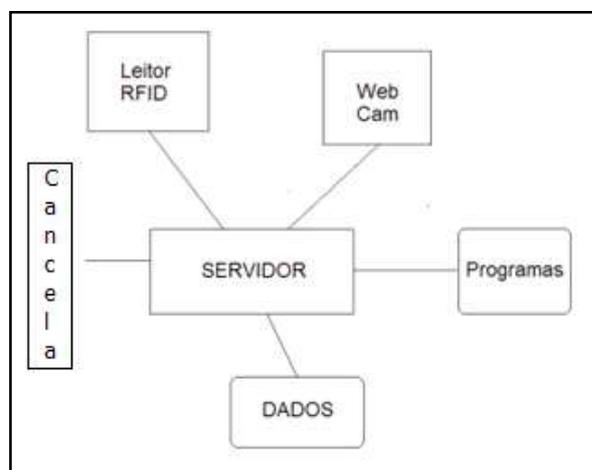


Figura10: Componentes Básicos do Sistema *Condominium*

A opção para demonstração do projeto consiste em duas máquinas representadas pelo servidor e pelo cliente junto com a leitora, conectadas a um switch de 8 portas. Além disto os hardwares estão ligados em rede por um cabo par trançado RJ 45.

3.2 Especificações do Projeto

Para realizar a implementação do software “Condominium” optou-se pelo uso das seguintes ferramentas de software livre:

- Ferramenta de modelagem Power Design [10]
- Sistema de gerenciador de banco de dados Mysql [11]
- Ferramenta de desenvolvimento para web PHP [12]
- Servidor de aplicação Tomcat [13]
- Máquina virtual Java [14]
- Ferramenta de desenvolvimento NetBeans [15]

Esta organização pode ser observada no organograma apresentado na Figura

11

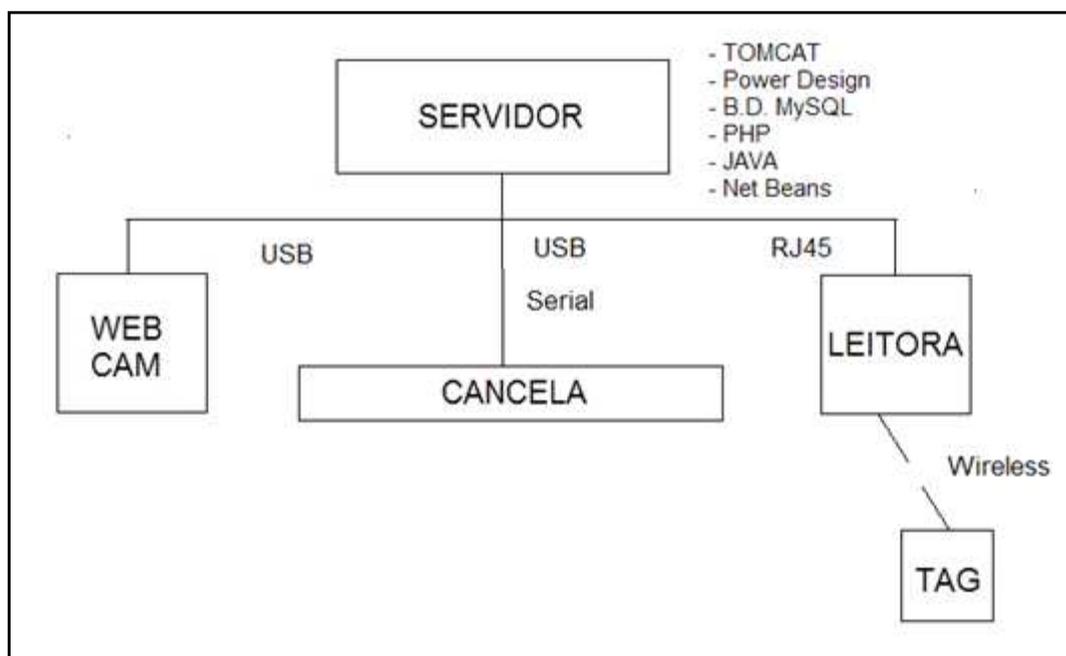


Figura 11: Organograma do Sistema *Condominium*

3.3 Hardware

A escolha do hardware para a implementação do Sistema *Condominium* primou na interoperabilidade de cada um dos seus componentes. Quanto mais simples e direta a proposta de hardware para atender às necessidades de segurança e acesso de veículos no interior do condomínio, mais fáceis serão os procedimentos de teste e implementação final.

Ao se utilizar componentes que trabalham com protocolos padrão da indústria, garante-se um alto nível de compatibilidade entre os mesmos, sendo possível a substituição de modelos com pouca ou nenhuma necessidade de recodificação dos applets responsáveis pela troca de informações entre os dispositivos.

3.3.1 Servidor

A configuração mínima do servidor para gerenciar todas as funcionalidades do Sistema *Condominium* requer: Processador com clock de Clock 1,66 Ghz; 1 Gb memória RAM; Disco rígido de 160 Gb e Sistema operacional Windows 7

O servidor utilizado no protótipo deste sistema foi um Netbook da marca Acer, modelo AOD250-1908, conforme visualizado na Figura 12.



Figura 12: Netbook Acer – Modelo AOD250-1908 [20]

3.3.2 Leitora

A leitora ideal para ser utilizada no sistema tem que ser capaz de gerenciar a rede e oferecer recursos de segurança. Outras características consideradas essenciais são a compatibilidade com o padrão DHCP e firmware e segurança baseada em SSL/SSH.

No protótipo do Sistema *Condominium* fez-se a escolha pela Leitora UHF Astra, ilustrada na Figura 13, devido a sua capacidade de ser alimentada pela rede ethernet ou DC sobre Wi-Fi.



Figura 13: Leitora UHF Astra [21]

3.3.3 Web Cam

A webcam ideal para ser utilizada no sistema tem que ser capaz de trabalhar com qualquer programa de captura e envio de imagem compatível com o Sistema Operacional Windows 7. Sua resolução mínima obrigatória é VGA, ou 640 x 480 pixels.

A webcam escolhida para o protótipo do Sistema *Condominium* foi a Webcam Creative Nx pro, visualizada na Figura 14, tanto pela compatibilidade com o Sistema Windows 7 quanto pela sua leveza e capacidade de ser facilmente encaixada em qualquer superfície.



Figura 14: Webcam Creative Nx pro [22]

3.3.4 Cancela

O modelo de cancela ideal para o Sistema Condominiumtem que obedecer aos padrões do mercado, ser flexível e oferecer uma boa relação custo-benefício. Entre os conectores desejados deverão estar as portas seriais do tipo RS232.

O modelo selecionado foi o TCOMPORT Corversor Multipoint e a Placa de 8 relês Multipoint, que podem ser vistos nas Figuras 15 e 16, por oferecer todas as funcionalidades necessárias para o protótipo.



Figura 15: TCOMPORT Corversor Multipoint

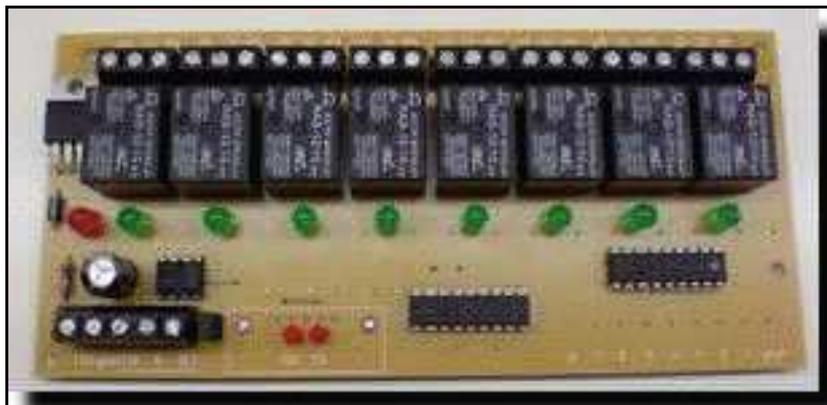


Figura 16: Placa de 8 relês Multipoint

3.4 Integração do Sistema

A integração de sistema está relacionada com os procedimentos para identificar e analisar as necessidades de um determinado ambiente tecnológico de uma organização e procurar implementar melhor solução para atendê-las em alguns pontos principais, como, do investimento, da segurança, da disponibilidade e eficiência. [17]

Os sistemas RFID podem ser criados com os mais diversos níveis de requerimento de hardware e software. Os desenvolvedores já encontram no mercado ferramentas capazes de superar diversos obstáculos a sua integração. Deve-se também ressaltar que, as aplicações RFID por mais que sejam beneficiados por lógicas modulares presentes nas ferramentas de integração, podem necessitar que algum código seja criado para permitir e aprimorar a eficiência da troca de informação entre os diversos aparelhos. [18]

Entre as funcionalidades dos sistemas RFID estão a leitura de mais de uma tag; da filtragem e agregação de dados; poder ler e escrever novas informações nas tags (para tags ativas); a integração dos sistemas às infraestruturas existentes; e da aplicação de regras, normas e leis ao acesso às informações. [18]

Considerando o conceito acima, pode-se ressaltar a três tipos de integração: de aplicações, de dados e de sistemas legados.

O Sistema *Condominium* aborda o conceito da integração de aplicações, uma vez que procura integrar diversas aplicações existentes (hardware e software) de modo que possam compartilhar informações.[18]

3.4.1 O Módulo de Detecção

Ele é constituído por um conjunto de web cams localizado nas zonas de acesso do condomínio e dos leitores de Tag passivas. Os locais de instalação devem ser cuidadosamente escolhidos para que o condômino, dentro do seu veículo, seja capaz de encontrá-la com facilidade. Cada uma das zonas de acesso deverá ter, no mínimo, uma web cam. Neste projeto utilizou-se uma web cam por zona de acesso.

Próximo as web cams está o leitor de Tags passivas. Este aparelho capta o código de identificação das diversas Tags passivas que passam no seu raio de alcance, e não apenas aquelas cadastradas no subsistema central.

A Tag passiva disponibilizada a cada um dos veículos dos condôminos após seu cadastramento no subsistema central, ao chegar ao raio de ação do leitor, ativa o protocolo de comunicação enviando seu código de identificação único.

3.4.2 O Programa Central

A Figura 17 oferece uma visualização geral do programa central e funcionalidades.

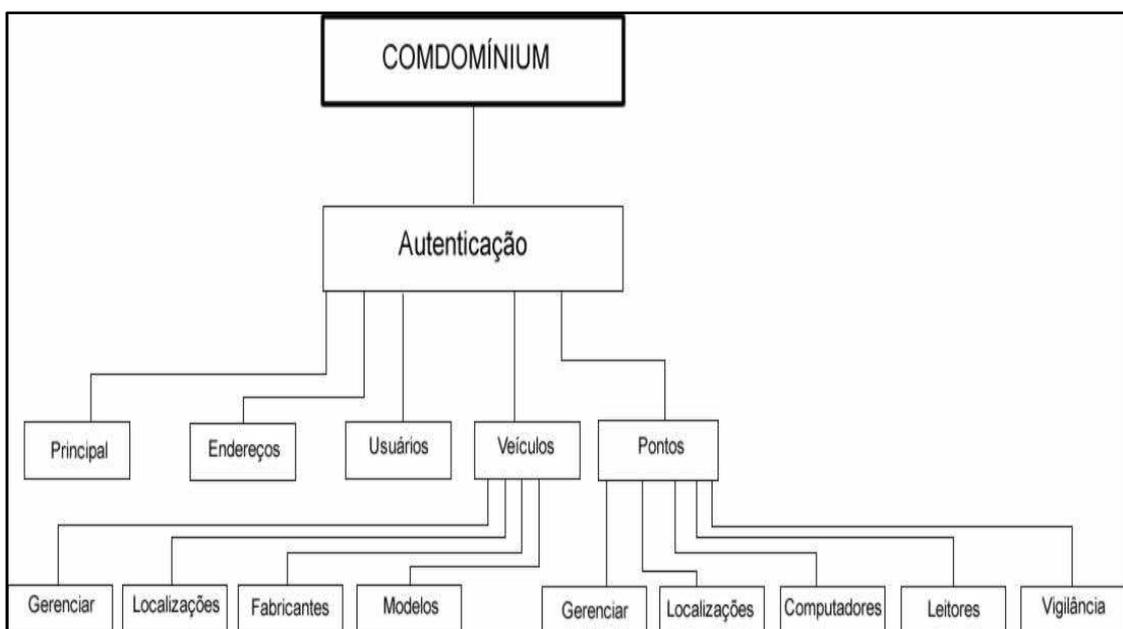


Figura 17: Fluxograma funcionalidades do Sistema *Condominium*

O programa central é o programa principal que controla a todos os outros subsistemas instalados. Além disso, ele controla o banco de dados do sistema de controle de acesso ao condomínio e registra num histórico de ocorrências todos os movimentos de entrada e saída. Ele reconhece cada uma das Tags passivas cadastradas e seus proprietários.

O programa central fica instalado na estação de trabalho principal localizada na portaria e que é utilizado pelo porteiro. No momento que for inicializado, considera-se que o leitor de Tags não realizará nenhuma leitura, pois não está sendo considerada a possibilidade de identificar todo e qualquer veículo cadastrado no interior do condomínio. Deve-se ressaltar que o sistema de controle de acesso ao condomínio tem a função de controlar o acesso e a saída de veículos, não o de controlar a sua presença no interior do condomínio.

Outro elemento considerado igualmente simples são as telas de computador já disponíveis na portaria do condomínio, que mostra a imagem de cada uma das web cams assim como as informações previamente gravadas no banco de dados do sistema sobre o condômino proprietário do veículo.

O porteiro visualiza na tela do computador todas as informações necessárias para a identificação do condômino proprietário do veículo e de sua família. A informação mostrada na tela do computador inclui além da informação do bloco e apartamento, fotografias de cada um dos moradores. Essas informações mostradas ao porteiro são controladas pelo subsistema central. Nele estão armazenadas as informações de todos os moradores do condomínio.

O programa central registra em histórico todas as leituras bem sucedidas das Tags pelos leitores, assim como do subsequente acesso as informações dos condôminos no banco de dados. Para uma maior segurança do condômino, o processo se repete tanto para a entrada do veículo no condomínio quanto para o momento da sua saída. O evento da entrada ou saída fica registrado no banco de dados quando o porteiro, manualmente, ativa o levantamento da cancela.

Além do relatado acima, o programa central também controla os procedimentos de manutenção da base de dados, incluindo seu backup. Além de controlar e gravar os registros históricos de entrada e saída, o subsistema central disponibiliza ao administrador do condomínio e condôminos, relatórios de controle de acesso. O administrador também tem a sua disposição relatórios específicos para

facilitar uma melhor gestão da escala de porteiros, para agilizar o processo de identificação e autorização nos períodos do dia de maior movimento.

3.4.3 O Módulo Gerencial

O módulo gerencial é responsável pelo cadastramento dos condôminos, veículos e Tags passivas na base de dados, assim como da sua manutenção. Somente o administrador do condomínio terá a autoridade para atualizá-los.

3.5 Operação do Sistema

A primeira parte do sistema trata do cadastramento de todos os condôminos e dos seus veículos e da atribuição de uma tag passiva para cada um dos veículos. A segunda parte é a responsável pela monitoração e controle de acesso dos veículos ao interior do condomínio. A terceira parte do sistema trata dos informes de entrada e saída dos veículos para consulta dos condôminos e pelo administrador do condomínio.

3.5.1 Cadastramento

Um profissional devidamente autorizado pelo administrador do condomínio realizará o cadastramento dos moradores e de seus veículos. Cada um dos condôminos deve tirar ou disponibilizar uma fotografia digital de seu rosto para completar o cadastro. As informações reunidas são carregadas num banco de dados MySQL, onde serão relacionadas com o número do apartamento e com seus moradores e veículos.

Uma vez finalizada a carga de dados, a cada um dos veículos é assignado uma tag passiva, que deverá ficar fixa no interior do mesmo. Essa Tag passiva é responsável pela identificação do veiculo e permite que o sistema possa recuperar as informações de seus proprietários.

3.5.2 Monitoração e Controle do Acesso de Veículos

3.5.2.1 Controle da Entrada de Veículos

Os leitores UHF, fixados em locais próprios próximo dos pontos de acesso a área do condomínio, identificam os automóveis cadastrados para ingressar no interior do condomínio.

Uma vez no alcance do leitor, a tag passiva presente no interior do automóvel responde ao pulso do leitor UHF. Uma vez realizada a leitura da tag passiva, o leitor UHF, por meio de uma classe Java, transmite o código de identificação do veículo para o servidor do condomínio. Recebida a informação, o sistema de monitoramento e controle faz acesso ao banco de dados MySQL para recuperar as informações do apartamento ao qual pertence o veículo.

No caso de uma identificação positiva, o sistema envia uma mensagem ao porteiro responsável pelo controle físico de entrada de veículos que além de incluir o número do apartamento, também disponibiliza as fotografias dos moradores. Caso seja negativa, o sistema informa que o veículo não está cadastrado para acessar o condomínio.

Ao chegar à cancela de acesso ao condomínio, o motorista do veículo deve posicionar o seu rosto em frente de uma web cam de segurança para que o porteiro possa fazer o seu reconhecimento visual, e, conforme o caso, permitir o acesso do veículo, apertando um botão pela elevação da cancela.

No momento em que o porteiro autoriza a entrada do veículo, ao apertar o botão da cancela, o sistema de monitoração de controle de acesso, por meio de outra classe Java, grava um registro no histórico de entradas e saídas com as informações pertinentes à entrada, ou seja, da identificação do veículo e data e hora. Opcionalmente, pode-se gravar, também, uma fotografia digital do veículo ingressando na área comum do condomínio.

3.5.2.2 Controle da Saída de Veículos

Assim como para o controle de entrada, haverá leitores de UHF e web cams fotográficas nos pontos de saída de veículos do condomínio. O procedimento de saída é similar ao de entrada, ou seja, o veículo para sair da área comum do condomínio passará pelo raio de alcance de um leitor UHF que, por meio de uma

classe Java, transmitirá ao sistema de monitoramento e controle os dados do apartamento e de seus moradores, para que o porteiro possa fazer uma identificação visual do motorista e liberar sua saída, abrindo a cancela.

Ao abrir a cancela, assim como no procedimento de entrada, por meio de uma classe Java, será gravado um registro no histórico de entradas e saídas, contendo as informações do veículo e apartamento, assim como data e hora.

3.5.3 Relatórios de Controle

Conforme relatado anteriormente, o programa principal do sistema de monitoramento e controle registra em arquivo histórico, todas as entradas e saídas dos veículos cadastrados no sistema. Essa fonte de informação pode ser utilizada tanto pelos moradores do condomínio quanto pela administração do condomínio para realizar um melhor controle das ocorrências.

Os condôminos têm acesso às informações relacionadas com os veículos do seu apartamento, sendo vedado o acesso às informações de terceiros.

O administrador do condomínio tem a sua disposição outros tipos de relatórios, voltados a uma melhor administração de tempos e movimentos de veículos e da conseqüente alocação de recursos humanos para a devida autorização de entrada e saída de veículos.

O administrador, de posse das informações de entrada e saída de cada um dos veículos, pode utilizar-se delas para uma série de eventos onde a sua atuação é colocada em dúvida, como, por exemplo, nos casos de acidentes que venham a danificar os veículos no interior do condomínio e de roubo de objetos que estavam no interior do veículo.

O sistema de monitoração e controle de acesso para condomínios não oferece apenas um controle duplo de acesso de veículos ao interior do condomínio, mas, também, a monitoração de sua presença, facilitando, assim, a ação de defesa do condomínio quando das denúncias de danos causados ao veículo nas áreas interiores.

3.6 Protótipo

Neste item tem-se uma breve descrição do protótipo desenvolvido para demonstrar o funcionamento do sistema.

3.6.1 A Infraestrutura de Manutenção de Dados.

A infraestrutura do programa principal foi construída com as ferramentas Netbeans e a linguagem Java para web. O banco de dados para armazenar e gerenciar os dados foi desenvolvido com a tecnologia MySQL. O Tomcaté utilizado como servidor web. Todos os softwares utilizados são de código aberto.

Inicialmente, faz-se o acesso ao sistema. Em seguida, cadastram-se no banco de dados as informações de alguns condôminos e seus veículos. Posteriormente cria-se o relacionamento do veículo com a Tag passiva. Todas estas informações ficam armazenadas no banco de dados MySQL.

A tela de “Login” representada na Figura 18 disponibiliza ao usuário os campos necessários para sua identificação e informação de senha de segurança.



Figura18: Tela de Login

Uma vez confirmada a sua identificação, o usuário administrador poderá cadastrar ou atualizar os dados dos usuários do Sistema *Condominium*. A Figura 19 apresenta a tela de manutenção dos dados de usuários do Sistema.

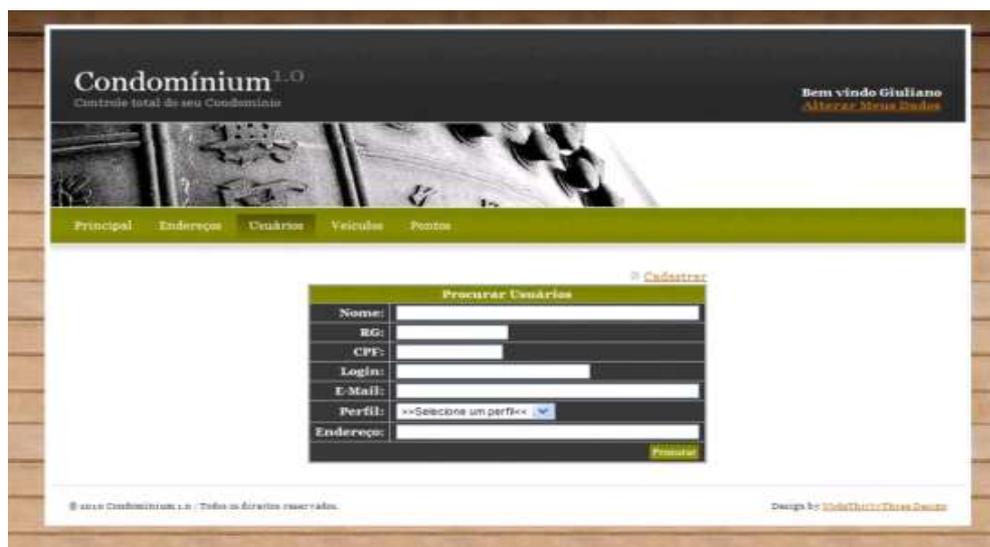


Figura 19: Tela Usuários

A Figura 20 oferece uma visão das informações necessárias para o cadastramento de veículos no Sistema *Condominium*.

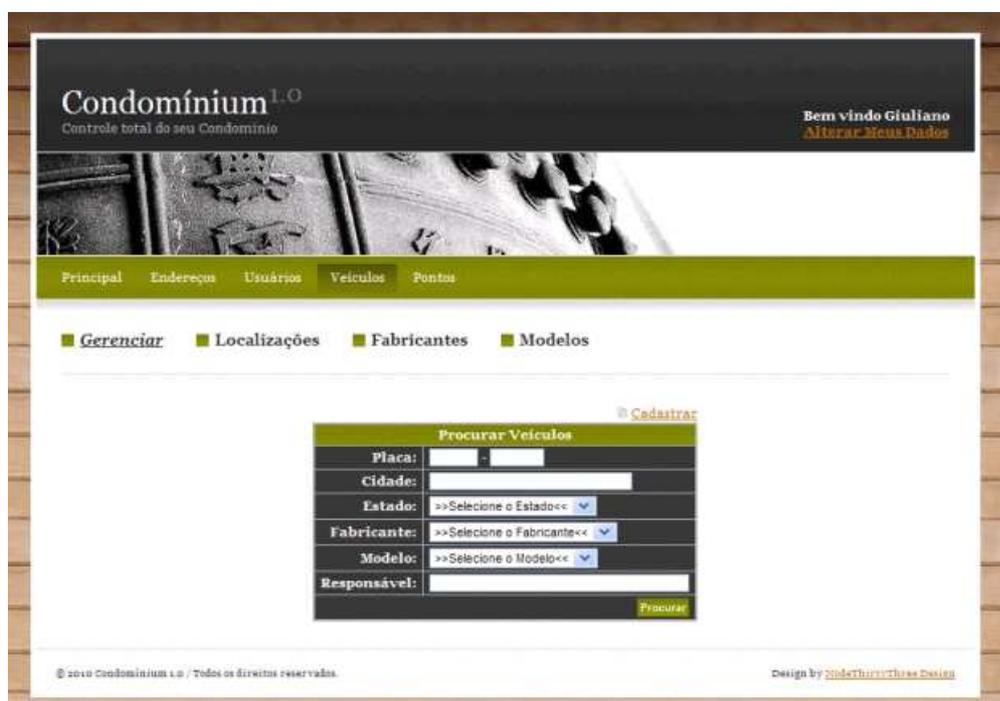


Figura 20: Tela Veículos

3.6.1 Funcionamento do Protótipo

A parte física ou de hardware do protótipo representa uma zona de acesso ao condomínio, apresentando todos os instrumentos de segurança, inclusive, a cancela. A figura a seguir possibilita um melhor entendimento deste esquema. A zona de acesso simulada tem apenas uma zona de transição, ou seja, de fora para dentro, no momento de entrar no condomínio, ou de dentro para fora, quando do momento da saída.

A Figura 21 disponibiliza uma visão geral do cenário.

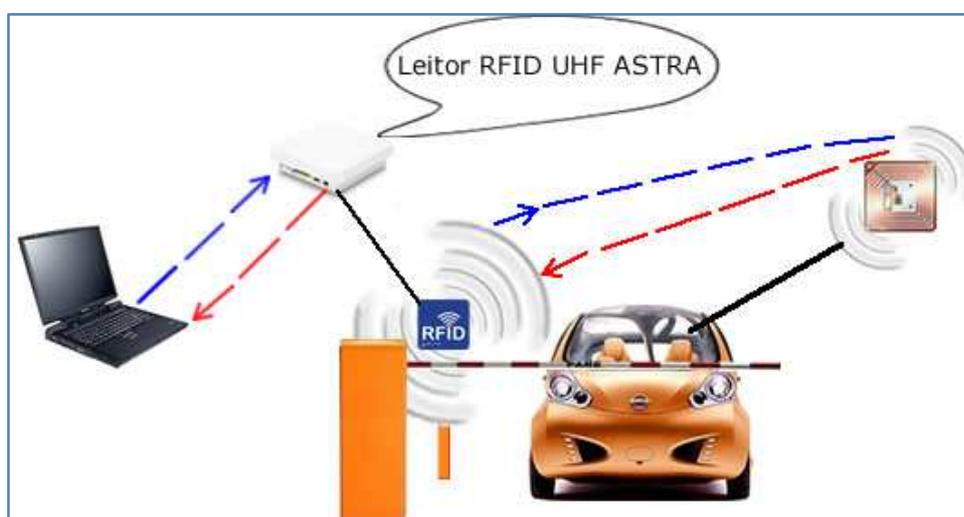


Figura21: Protótipo

São apreciadas duas situações, do reconhecimento ou do não-reconhecimento da Tag passiva pelo leitor.

O subsistema central é o único programa que interage entre o leitor/web cam, a estação de trabalho na portaria e a cancela. Ele funciona como programa principal responsável pela comunicação com os outros subsistemas e atualização do banco de dados.

A circunferência de leitura do leitor RFID foi aqui reproduzido neste modelo de modo a reproduzir uma zona de acesso. Para a construção deste protótipo foi utilizado o seguinte hardware:

- a) Um leitor RFID – UHF
- b) Duas Tags passivas – uma cadastrada e outra, não cadastrada.

- c) Uma web cam;
- d) Um netbook; e
- e) Uma Cancela.

A operação do protótipo foi realizada da forma mais real possível. Foi simulado o cadastramento de um veículo e da respectiva Tag utilizando o programa principal. Neste ponto o programa comunica ao operador que o veículo foi devidamente registrado. Em seguida, têm-se a simulação da tentativa de acesso do mesmo veículo ao interior do condomínio. O subsistema central foi informado corretamente pelo protocolo do leitor dos dados da Tag.

O subsistema central acessa o banco de dados e disponibiliza na tela do netbook a fotografia do condômino, conforme pode ser observado nas Figuras 22 e 23. O protocolo da web cam também envia para o netbook as imagens que estão sendo captadas. Dessa forma o porteiro poderá autorizar ou não o acesso ao interior do condomínio.

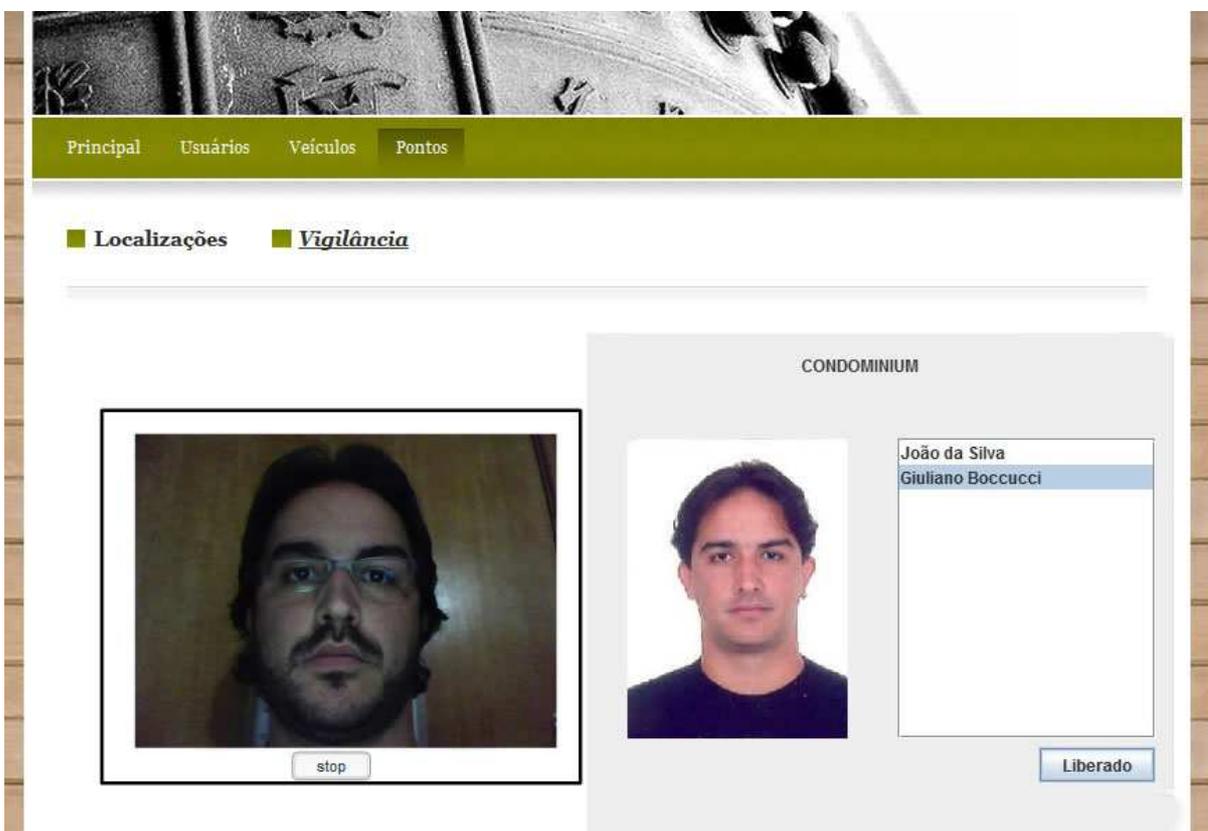


Figura 22: Tela de Confirmação de Entrada ou Saída

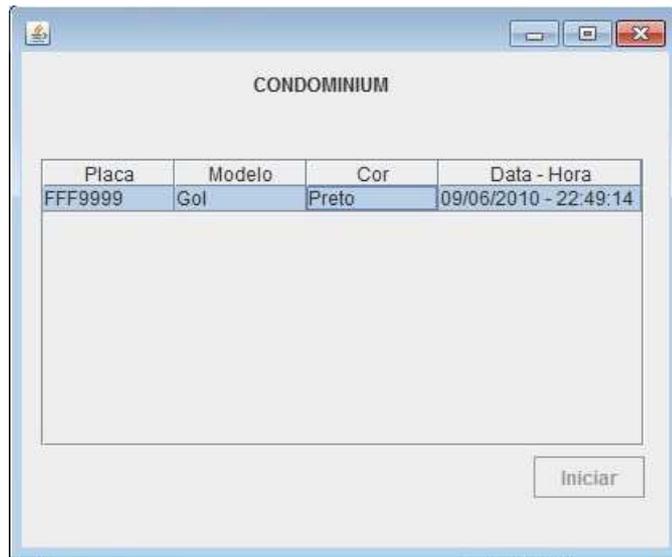


Figura 23: Tela de Confirmação de Entrada ou Saída

A ativação da cancela(Figuras 24, 25 e 26) é providenciada pelo programa central executado no netbook, que a ele se conecta por meio de uma porta serial. Foi necessário desenvolver um novo protocolo para que tal comunicação ocorresse de forma ótima para o sistema. Foi igualmente necessário instalar um adaptador para conectar a cancela ao computador utilizando a porta USB.

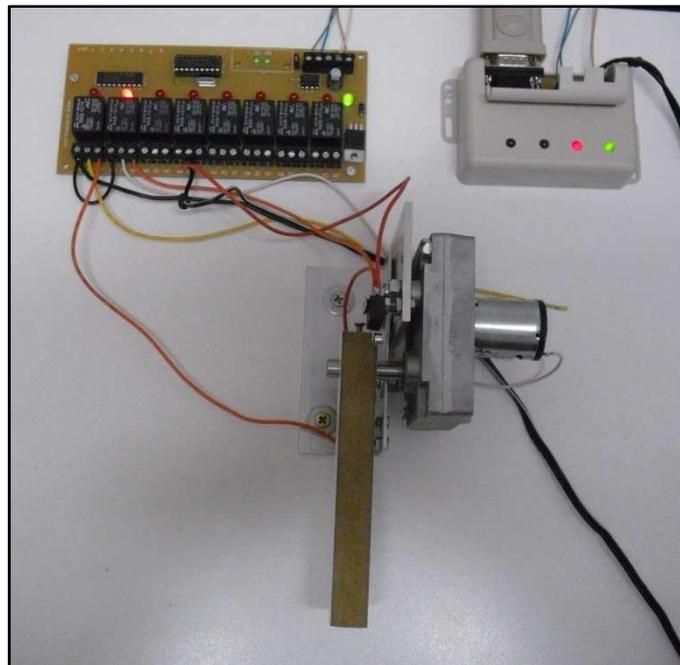


Figura 24 - Cancela

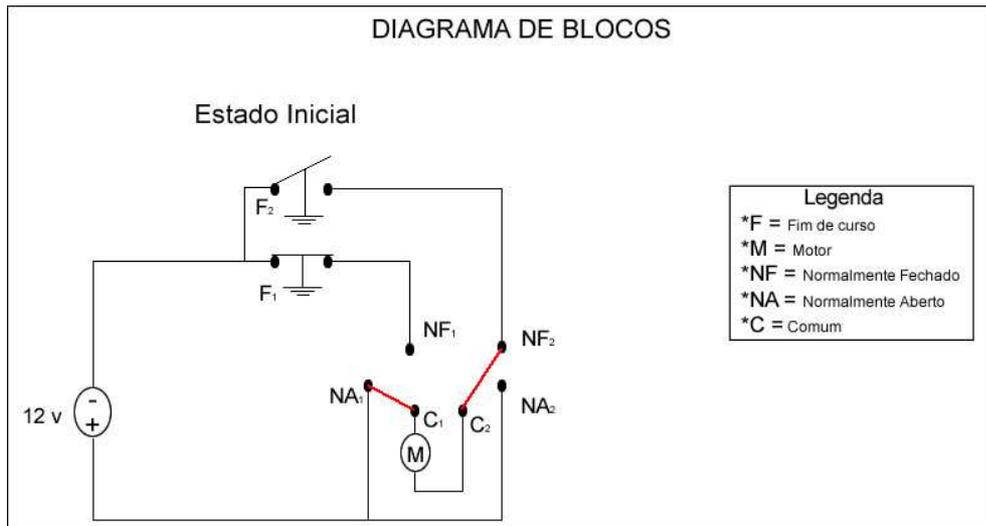


Figura 25 – Funcionamento da cancela

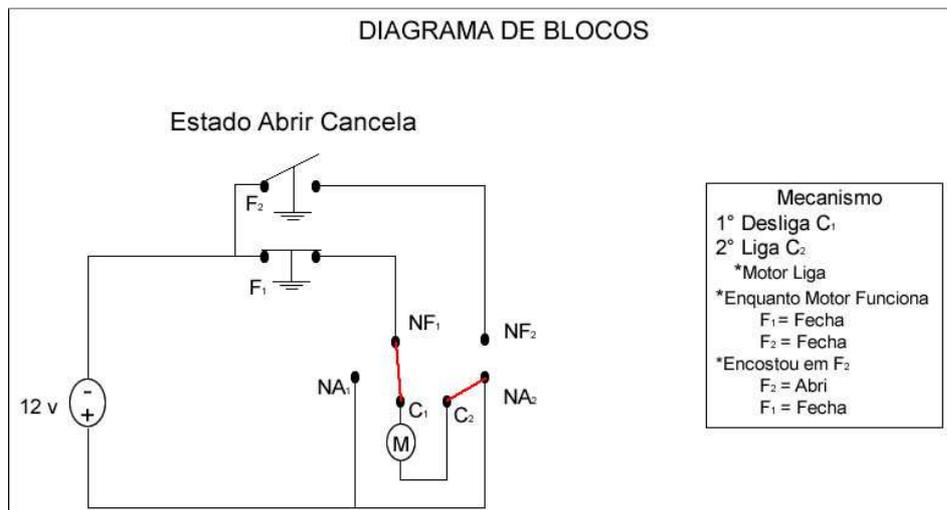


Figura 26 – Funcionamento da cancela

Os relatórios à disposição dos condôminos e da administração possibilitam o controle de entrada e saída de veículos. Eles também disponibilizam as informações do proprietário do veículo (Figura 27).

Condomínium 1.0
Controle total do seu Condomínio

Bem vindo Administrador
Alterar Meus Dados
Sair

Principal Endereços Usuários Veículos Pontos

Gerenciar Localizações Fabricantes Modelos

Registro de Localizações de um veículo

Placa:

Fabricante: >> Todos << ▼

Modelo: >> Selecione o Modelo << ▼

Usuário:

Data Inicial:

Data Final:

Procurar

Data Hora	Ponto	Situação	Fabricante	Modelo	Placa	Cor	Motociclista
11/06/2010 - 08:20	Entrada Lateral A	Saída	Honda	Civic	JJ18888	Azul	João
12/05/2010 - 15:17	Entrada Lateral A	Saída	Honda	Civic	JJ18888	Azul	João
04/04/2010 - 11:44	teste Ponto	Entrada	Honda	Civic	JJ18888	Azul	Giuliano
12/03/2010 - 20:32	teste Ponto	Entrada	Fiat	Stilo	JJ5555	preto	Juliana

Com os dados fornecidos acima, existem 4 localizações

Figura 27: Relatório de Entrada e Saída de Veículos

A versão final do protótipo funcionou corretamente para as possibilidades planejadas. Considera-se possível oferecer novas funcionalidades ao sistema de controle de acesso ao condomínio num momento futuro.

4 CONCLUSÃO

A utilização das técnicas de integração de sistema demonstrou ser positiva para o desenvolvimento do Sistema *Condominium*. Além de oferecer uma abordagem modular, a construção do projeto permite uma adequação eficiente dos equipamentos novos com aqueles presentes, cuja intenção maior é a de atender as necessidades da organização, ou seja, da garantia de retorno do investimento, da disponibilidade com segurança das informações desejadas e, principalmente, da eficiência na integração de todos os componentes.

O mercado nacional tem a disposição as mais variadas plataformas para o desenvolvimento de sistemas integrados, direcionados ao aproveitamento da tecnologia RFID. Eles compartilham entre si um objetivo comum: de superar as limitações de comunicação entre os mais diversos tipos de componentes.

A tecnologia RFID está capacitada a ser utilizada de forma modular assim como suas funcionalidades maiores. Entre elas estão a possibilidade de maior adaptação a infraestruturas existentes, leitura concorrente de mais de uma tag, da seleção de quais informações serão transmitidas entre os diversos componentes, e também da aplicação de regras, normas e leis ao acesso às informações, garantindo, assim, a privacidade de seus usuários. O Sistema *Condominium* procurou tirar proveito de toda essa flexibilidade.

O sistema e o protótipo aqui apresentados são bastante simples, havendo, assim, muitas oportunidades para aprimorá-lo. Ele funciona satisfatoriamente para o que se propôs, de controlar o acesso de veículos. Este sistema pode ser constituído por diversas circunferências de detecção RFID, cada uma com o melhor alcance para sua posição.

O sistema de segurança idealizado, embora simples, é suficiente para garantir aos condôminos a segurança de seus dados pessoais dos outros condôminos. Apenas o administrador do sistema tem acesso aos relatórios mais detalhados de entrada e saída de veículos.

Sobre o modelo do sistema de acesso, ele foi planejado para um cenário particular e controlado. Em casos mais complexos de formas de acesso e controle, devido à simplicidade de sua concepção, o sistema poderá ser adaptado conforme as necessidades que se apresentarem.

O hardware utilizado nesses sistemas obedece a padrões internacionais da indústria. Isto garante que as suas atualizações ocorram com poucos ou nenhum conflito.

Os Sistemas RFID não necessitam ser complexos para serem funcionais. O Exemplo apresentado neste trabalho, de um sistema integrado para controlar e registrar a entrada e saída de veículos comprova que as ferramentas utilizadas têm um elevado nível de compatibilidade, e quando foi necessário intercambiar seus elementos, os *applets* desenvolvidos para a troca de informações não necessitaram ser modificados, garantindo, assim a sua funcionalidade.

Este sistema aqui apresentado pode ser aprimorado com novas funcionalidades. Entre elas pode-se citar a inclusão de módulos relacionados com a biometria, ou seja, do uso de características biológicas em mecanismos de identificação. A identificação biométrica se mostra interessante porque é uma tecnologia que diminui a possibilidade de enganar o sistema, já que é difícil copiar uma característica física. No caso específico do Sistema *Condominium*, as possibilidades adicionais mais factíveis são do reconhecimento de voz e do reconhecimento dos traços do rosto.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] <http://nexus.eng.br/auto.html>. Acessado em 20/03/2010
- [2] <http://www.acura.com.br/> Acessado em 04/02/2010
- [3] <http://pt.wikipedia.org/wiki/RFID>. Acessado em 01/03/2010
- [4] SCAVARDA, L. F., FILHO, C. N., KRAEMER, V. RFID na Logística: fundamentos e Aplicações. ABEPRO, ENGEPP 2005.
- [5] BERNARDO, C. G. A tecnologia RFID e os benefícios da etiqueta inteligente para os negócios. Revista Eletrônica Unibero de Iniciação Científica, São Paulo, 2004.
- [6] CUNHA, Alessandro F. RFID - Etiquetas com eletrônica de ponta. SABER ELETRÔNICA. Nº 401. Junho, 2006.
- [7] NETO, Zamith França. Aplicação da tecnologia RFID para a gestão acadêmica, de pessoal e operacional de patrimônio e biblioteca. SPOLM 2008. ISSN 2175-6295. Rio de Janeiro- Brasil, 05 e 06 de agosto de 2008.
- [8] DYSON, Esther; DEAN, Eric. RFID: Logistics Meets Identity. Esther Dyson's Monthly Report. Vol. 21. No.6. June 2003.
- [9] BILLINGSLEY, K. Lloyd. Playing Tag: An RFID Primer. Pacific Research Institute. San Francisco, CA. July, 2007.
- [10] <http://www.sysbase.com.br> Acessado em 14/03/2010
- [11] <http://www.mysql.com.br> Acessado em 14/03/2010
- [12] <http://www.baixaki.com.br/download/php.htm> Acessado em 14/03/2010
- [13] <http://tomcat.apache.org> Acessado em 14/03/2010
- [14] <http://developers.sun.com/downloads/top.jsp> Acessado em 14/03/2010
- [15] <http://netbeans.org> Acessado em 14/03/2010
- [16] <http://www.adobe.com> Acessado em 14/03/2010
- [17] <http://java.sun.com/javase/6/docs/tutorial/doc/> Acessado em 24/05/2010
- [18] <http://www-di.inf.puc-rio.br>. Acessado em 06 maio 2010
- [19] http://www.rfidsystems.com.br/listar_produtos.php?grupo=EPC_leitor_fixo&lista=epc. Acessado em 10 junho 2010

[20]<https://www.kabum.com.br/cgi-local/kabum3/produtos/descricao.cgi?id=01:04:03:85:207>. Acessado em 10 junho 2010

[21]http://www.google.com.br/url?sa=t&source=web&cd=2&ved=0CCEQFjAB&url=http%3A%2F%2Fwww.rfidsystems.com.br%2Flistar_produtos.php%3Fgrupo%3DEPC_leitor_fixo%26lista%3Depec&ei=2j8VTOWLK82PuAfh3JWMCg&usg=AFQjCNEJaY63FOKpEABpXEyAJDtceYSfvQ. Acessado em 10 junho 2010

[22]http://img.ibiubi.com.br/produtos/6/8/7/0/1/6/2/img/01_webcam-creative-labs-modelo-nx-pro_grande.jpg. Acessado em 10 junho 2010

Anexos

Anexo I – Características Técnicas da Leitora UHF Astra



<http://rfid.thingmagic.com/Portals/42741/images/ThingMagic-Astra25.jpg>

Arquitetura

- Processador: Intel IXP420 Network
- Memória DRAM: 64Mb
- Memória Flash: 32Mb
- Tag Buffer: 65.000 tags
- Sistema operacional: Linux

Recursos do MercuryOS

- Segurança: baseada em SSL/SSH
- Aplicações baseadas na web: configurações e monitoramentorealizados via browser; HTTP/HTTPS
- Comunicação: TCP/IP ou Wi-Fi

- Capacidade de executar internamente softwares para
- Gerenciamento de tags
- Trigger externo para leitura/gravação de tags

Desempenho

- Taxa de leitura: até 190 tags/segundo
- Distância de leitura: até 9 metros com a antena embutida de 6dBi (36 dBm EIRP)
- Capacidade de identificação EPC: 256 bits

Protocolo

- Protocolo de interface RFID: EPC Global Gen 2 (ISO 18000-6C) com anti-colisão, DRM.

Aspecto Físico

- Dimensões (L x A x P): 260 x 260 x 76 mm

Interface RFID

- Conector da antena externa: 1 RP TNC para suportar antena externa mono-estática, além da antena integrada
- Potência de saída: níveis de leitura e gravação separados e configuráveis, de 5 dBm a 30 dBm (1W), precisão de ± 1.0 dBm
- Frequência de operação: 902-907 MHz, 915-928 MHz

Interface de controle de dados

- Conectores: interface Ethernet 10/100 Base-T; porta serial RS-232; terminal de 12 pinos para conexão por parafuso

- Wireless: 802.11 b/g (opcional); chaves WEP 40-bit e 104-bit; WPA e WPA2, com algoritmos TKIP e AES com chaves pré-compartilhadas ou EAP-TLS
- Indicadores: 1 LED indicador

Anexo II - Creative Webcam NX Pro USB

Image Resolution: 1024 x 768

Video Resolution: 640 x 480 @ 15 fps

Sensor: CMOS

Interfaces: USB

Camera Type: Webcam

Tilt/Swivel: 15° Up / 30° Down Tilt ; 360° Swivel

Lens

Focus Mode: Manual

Lens Type: Manual focus lens

Physical Characteristics

Dimensions: 3" Height x 10" Width x 12" Depth

Miscellaneous

Additional Information

Dedicated microphone for superior audio

Works with Yahoo Messenger, Windows Messenger, and virtually all Windows programs that support video

Snapshot button

Three-way base sits on top of your PC or CRT monitor, and attaches to most flat panel and notebook PC displays

Anexo III - Netbook Acer AOD250-1908

Características:

- Windows® 7 Starter Autêntico
- Processador Intel® Atom™ N270 (512KB L2 cache, 1.66GHz, 533MHz FSB)
- Memória DDR2 533/667 MHz SDRAM memory interface design
- Tela 10.1" SD 1024 x 600 (WSVGA) pixel resolution, high-brightness (200-nit) Acer CrystalBrite™ LED-backlit TFT LCD
- Armazenamento Hard disk 2.5" 9.5 mm 160 GB
- Multi-in-1 card reader

Audio

Audio Suporte audio High-definition

Dois auto falantes Stereos

Compatível MS-Sound

Microfone embutido

Comunicação

Webcam integrada Acer Crystal Eye, suporte resolução 0.3 megapixel

WLAN: Acer InviLink™ 802.11b/g Wi-Fi CERTIFIED® network connection, suporte Acer SignalUp™ wireless technology

LAN: 10/100 Mbps Fast Ethernet

Dimensões e Peso

258.5mm (L) x 204mm (P) x 25.4mm (A)

1.11 kg para SKUs com bateria 3-cell pack

Energia

30 W adaptador com cabo de energia

Segurança de Produtos Elétricos e Materiais (PSE), certificados para bateria.

24.4 W 2200 mAh 3-cell Li-ion battery pack

- up to 3.15-hour battery life

Teclas especiais e controles

Teclado 84-/85-/88-teclas, 89% do tamanho de um teclado padrão, 1.6 mm (min) key travel

Multi-gesture touchpad com dois botões, suporte:

- Movimento Circular
- Zoom
- Rolamento de página

12 teclas de função, quatro teclas de cursor, uma tecla do Windows ® para Windows ® XP Home, controles hotkey, teclado numérico, suporte de idiomas internacionais

Botão lig/des com LED

Fácil acesso WLAN/WWAN* botão com LED

I/O ports

Três portas USB 2.0

Porta Externa (VGA)

Headphone/speaker/line-out jack

Microphone-in jack

Porta Ethernet (RJ-45)

DC-in jack para AC adaptador

Anexo IV - Placa de 8 relês Multipoint

Características da placa

- Instale junto outras placas e módulos Multipoint
- Acionamento por ID : torna imune contra acionamento indevido.
- ID distinto para placas iguais no mesmo cabo.
- Até 32 placas idênticas no mesmo cabo
- Retorno do estado atual da placa .
- Não muda o estado dos relês caso o computador seja reiniciado
- Cada placa pode ser instalada junto ao local onde as cargas estão (instale junto a disjuntores a até 1200 metros do computador)
- 8 saídas independentes (pode ligar cargas de voltagens diferentes em cada saída)
- Alimentação 12 volts

Identificação dos bornes de alimentação e controle

Borne 12v . Ligue positivo da fonte de 12volts

Borne GND Ligue negativo da fonte

Borne H Terra opcional entre placas (opcional)

Borne A .Sinal de controle A

Borne B . Sinal de controle B

Atenção:

Entre todas as placas Multipoint e o conversor Multipoint os sinais A ligam com A e B ligam com B .

Use fio de telefone ou cabo de rede CAT5 (cabo azul de rede)

Cada placa tem um strap ao lado do borne B . Somente na ultima placa do barramento o strap deve ficar fechado permanecendo todas as outras com strap aberto . No caso de apenas uma placa o strap deve ficar fechado .

O cabo de dados pode ter o comprimento máximo de 1200 metros ..

Antes de instalar faça um teste com um pequeno cabo

Para distâncias de mais de 10 metros a alimentação de 12 volts deve ser feita no local de cada placa .

Teste da placa

Use cabo extensor serial onde todos os pinos são ligados sem cruzamento de 1 a 1 até 9 a 9 .

O cabo vai conectado entre a porta COM de seu computador e o conversor Multipoint.Com a alimentação e cabo de dados ligado rode o programa de teste Multipoint .Escolha a porta COM física que seu computador dispõe e clique em conectar .Basta clicar nos botões virtuais e verá os leds da placa acender junto a um click sonoro do relê.A placa responde no painel ao lado com seu ID seguido do estado do relê .Ex Botão ligar 1 a placa responde 8OUAL1 onde 8OUA é o ID da placa , L de ligar e 1 de relê 1Ao clicar em desligar 1 a placa responde 8OUAD1 onde D de Desligar e 1 de relê um .Assim é possível ter certeza do funcionamento da placa a distância .

Demais placas no barramento respondem com IDs diferentes . A segunda placa de 8 relês do barramento possui o ID 8OUB .

Erros de configuração de porta COM

Caso apresente erro ao clicar em “conectar” é uma indicação que a porta COM física não é a mesma que o programa está tentando usar .

Certifique-se que a porta COM que selecionou no programa é a porta física .

Vá em Painel de controle / propriedades do sistema / hardware / gerenciador de dispositivos/portas COM e LPT . Dê dois cliques. Apareceram as portas disponíveis .

Cabo conversor USB serial

Use cabo conversor USB serial . Antes de conectar o cabo ao computador instale o driver que acompanha o cabo .

Normalmente o windows escolhe uma porta COM alta para colocar o cabo . O programa de controle só oferece até COM 4 .

Para saber onde em qual porta COM o cabo foi instalado vá em :Painel de controle / propriedades do sistema / hardware / gerenciador de dispositivos /portas COM e LPT . Dê dois cliques.Apareceram as portas disponíveis . Caso seja necessário mude a porta COM do cabo clicando com botão da direita na porta COM virtual que foi criada veja propriedades/configurações de porta /avançado. Em numero da porta COM escolha uma porta COM baixa 1 ,2, 3 ou 4 . Dê preferência a porta COM1 no caso de computadores onde não tem porta COM nativa. E porta COM2 onde já existe uma porta COM.

Salve e reinicie o computador.

Remover o cabo adaptador serial USB causa perda destas configurações.

APÊNDICE B - Scripts de criação do modelo de dados

```
drop table if exists DO_ESTADO;
drop table if exists DO_PERFIL;
drop table if exists DO_TIPOTELEFONE;
drop table if exists TA_VEI_USU;
drop table if exists TB_ANTENA;
drop table if exists TB_CANCELA;
drop table if exists TB_COMPUTADOR;
drop table if exists TB_ENDERECO;
drop table if exists TB_FABRICANTE;
drop table if exists TB_LEITOR;
drop table if exists TB_LOCALIZACAO;
drop table if exists TB_MODELO;
drop table if exists TB_PONTO;
drop table if exists TB_TELEFONE;
drop table if exists TB_USUARIO;
drop table if exists TB_VEICULO;
```

Tabela Estado

```
create table DO_ESTADO
(
  ESTID          char(2) not null comment 'Identificador da tabela estado. Campo obrigatório.',
  ESTNOME        varchar(30) not null comment 'Nome do estado. Campo obrigatório.',
  primary key (ESTID)
);
```

Tabela Perfil

```
create table DO_PERFIL
(
  PERID          int(2) unsigned not null auto_increment comment 'Identificador da tabela perfil. Campo obrigatório',
  PERNOME        varchar(50) not null comment 'Nome do perfil. Campo obrigatório.',
  primary key (PERID)
);
```

Tabela Tipo Telefone

```
create table DO_TIPOTELEFONE
(
  TIPID          int(2) not null auto_increment comment 'Identificador da tabela tipo de telefone. Campo obrigatório.',
  TIPNOME        varchar(15) not null comment 'Campo que dirá se será telefone móvel ou fixo. Campo obrigatório.',
  primary key (TIPID)
);
```

Tabela Veículo Usuário

```
create table TA_VEI_USU
(
  VUFK_VEIID     char(11) not null comment 'Número identificador do veiculo',
  VUFK_USUID     int(8) not null comment 'Número identificador do usuário',
  VURESPONSAVEL boolean not null comment 'Informa qual o usuário responsável pelo veiculo. Repare que é possível inserir mais de um usuário responsável.',
  primary key (VUFK_VEIID, VUFK_USUID)
);
```

Tabela Antena

```
create table TB_ANTENA
(
  ANTID          int(8) unsigned not null auto_increment comment 'Identificador da tabela antena. Esse campo é obrigatório.',
  ANTFK_LEIID    int(8) not null comment 'Chave estrangeira da tabela leitora. Determina qual leitora a antena pertence. Campo obrigatório, pois toda antena tem que ter uma leitora.',
  ANTFK_PONID    int(8) not null comment 'Chave estrangeira da tabela ponto. Determina qual ponto a antena pertence. Campo obrigatório, pois toda antena tem que ter um ponto',
  ANTNUMERO      int(1) not null comment 'Esse é o número da porta que a antena está conectada no leitor.',
  ANTTIPO        char(1) not null comment 'Esse campo define se a antena está apontada para a entrada (E) ou para a saída (S) do condomínio.',
  primary key (ANTID)
);
```

Tabela Cancela

```
create table TB_CANCELA
(
  CANID          int(8) unsigned not null auto_increment comment 'Identificador da tabela cancela. Campo obrigatório',
  CANFK_COMID    int(8) not null comment 'Chave estrangeira da tabela computador. Identifica em qual computador a cancela esta instalada. Campo obrigatório',
  CANPORTASERIAL int(2) not null comment 'Identifica em que porta serial no computador a cancela esta instalada. Campo obrigatório',
  CANTIPO        char(1) not null,
  primary key (CANID)
);
```

Tabela Computador

```
create table TB_COMPUTADOR
(
  COMID          int(8) unsigned not null auto_increment comment 'Identificador da tabela computador. Campo obrigatório',
  COMFK_PONID   int(8) not null comment 'Chave estrangeira da tabela local. Determina o local fixo em que o computador se encontra. Campo obrigatório.',
  COMNOME       varchar(100) not null comment 'Determina o nome do computador. Campo obrigatório',
  primary key (COMID)
);
```

Tabela Endereço

```
create table TB_ENDEREÇO
(
  ENDID          int(8) unsigned not null auto_increment comment 'Identificador da tabela endereço. Campo obrigatório.',
  ENDNUMERO     varchar(8) not null comment 'Número do endereço do morador. Campo obrigatório.',
  ENDBLOCO     varchar(5) comment 'Bloco do morador. Não é um campo obrigatório',
  ENDQUADRA    varchar(5) comment 'Quadra do morador não é um campo obrigatório',
  ENDCONJUNTO  varchar(5) comment 'Conjunto do endereço do morador. Campo não obrigatório.',
  primary key (ENDID)
);
```

Tabela Fabricante

```
create table TB_FABRICANTE
(
  FABID          int(8) unsigned not null auto_increment comment 'Identificador da tabela fabricante. Campo obrigatório.',
  FABNOME       varchar(50) not null comment 'Nome do fabricante. Campo obrigatório.',
  primary key (FABID)
);
```

Tabela Leitor

```
create table TB_LEITOR
(
  LEIID          int(8) unsigned not null auto_increment comment 'Identificador da tabela leitor. Campo obrigatório.',
  LEIIP         varchar(15) not null comment 'Campo para inserir o IP da leitora. Campo obrigatório.',
  LEIPORTA      char(4) not null comment 'Campo para inserir qual a porta de comunicação da leitora. Campo obrigatório.',
  primary key (LEIID)
);
```

Tabela Localização

```
create table TB_LOCALIZACAO
(
  LOCID          int(8) not null,
  LOCFK_USUID   int(8) not null comment 'Chave estrangeira da tabela usuario . Determina o nome do usuário. Campo obrigatório.',
  LOCFK_VEIID   char(11) not null comment 'Chave estrangeira da tabela veiculo. Determina o nome do veiculo. Campo obrigatório.',
  LOCFK_ANTID   int(8) not null comment 'Chave estrangeira da tabela antenna . Determina o numero e o tipo da antenna. Campo obrigatório.',
  LOCDATAHORA  datetime not null comment 'Determina a data e a hora que o veiculo saiu ou entrou no estacionamento. Campo obrigatório.',
  primary key (LOCID)
);
```

Tabela Modelo

```
create table TB_MODELO
(
  MODID          int(8) unsigned not null auto_increment comment 'Identificador da tabela modelo do veículo. Campo obrigatório.',
  MODFK_FABID   int(8) not null comment 'Chave estrangeira da tabela fabricante. Identifica o fabricante (marca) do veículo. Campo obrigatório.',
  MODNOME       varchar(50) not null comment 'Nome do modelo do veículo. Campo obrigatório.',
  primary key (MODID)
);
```

Tabela Ponto

```
create table TB_PONTO
(
  PONID          int(8) unsigned not null auto_increment comment 'Identificador da tabela Ponto. É um campo obrigatório',
  PONNOME       varchar(50) not null,
  primary key (PONID)
);
```

Tabela Telefone

```
create table TB_TELEFONE
(
  TELID          int(8) not null auto_increment comment 'Identificador da tabela telefone. Campo obrigatório.',
  TELFK_TIPID   int(2) not null comment 'Chave estrangeira da tabela tipo de telefone. Identifica tipo de telefone do usuário. Campo obrigatório.',
  TELFK_USUID   int(8) not null comment 'Chave estrangeira da tabela usuário. Determina o usuário. Campo obrigatório.',
  TELDDI        char(3) not null comment 'Código de ligação internacional. Campo obrigatório.',
  TELDDD        char(2) not null comment 'Código de ligação nacional. Campo obrigatório.',
);
```

```
    TELNUMERO      char(8) not null comment 'Número do telefone. Campo obrigatório.',
    TELRAMAL       varchar(6) comment 'Ramal do telefone. Campo não é obrigatório.',
primary key (TELID)
);
```

Tabela Usuário

```
create table TB_USUARIO
(
    USUID          int(8) unsigned not null auto_increment comment 'Identificador da tabela Usuário. Campo obrigatório.',
    USUFK_PERID    int(2) not null comment 'Chave estrangeira da tabela perfil. Determina o nome do perfil do usuário. Campo obrigatório.',
    USUFK_ENDID    int(8) not null comment 'Chave estrangeira da tabela apartamento. Identifica o endereço do usuário. Campo obrigatório.',
    USUNOME        varchar(40) not null comment 'Nome do usuário. Campo obrigatório.',
    USUSOBRENOME   varchar(40) not null comment 'Sobrenome do usuário. Campo obrigatório.',
    USUSEXO        char(1) not null comment 'Sexo do usuário. Campo obrigatório.
        M - Masculino
        F - Feminino',
    USURG          varchar(15) not null comment 'Carteira de identidade do usuário. Campo obrigatório.',
    USUCPF         char(11) not null comment 'Cpf do usuário. Campo obrigatório.',
    USUEMAIL       varchar(50) comment 'Correio eletrônico do usuário. Campo não é obrigatório.',
    USULOGIN       varchar(30) not null comment 'Login do usuário. Campo não é obrigatório.',
    USUSENHA       varchar(32) not null comment 'Campo para senha de entrada no sistema, para cada usuário. Campo não é obrigatório.',
    USUATIVO       boolean not null comment 'Campo que irá informar se o usuário está ativo ou não no sistema. Campo obrigatório.',
    USURESPONSAVEL boolean not null,
primary key (USUID)
);
```

Tabela Veículo

```
create table TB_VEICULO
(
    VEIID          char(11) not null comment 'Identificador da tabela veículo. Campo obrigatório.',
    VEIFK_MODID    int(8) not null comment 'Chave estrangeira da tabela modelo. Determina qual o modelo do veículo. Campo obrigatório.',
    VEIFK_ESTID    char(2) not null comment 'Chave estrangeira da tabela estado. Determina qual o estado do veículo. Campo obrigatório.',
    VEIFK_ENDID    int(8) not null comment 'Identificador da tabela endereço. Campo obrigatório.',
    VEIPLACA       char(7) not null comment 'Placa do veículo. Campo obrigatório.',
    VEICIDADE      varchar(30) not null comment 'Cidade do veículo. Campo obrigatório.',
    VEICOR         varchar(15) not null comment 'Cor do veículo. Campo obrigatório.',
    VEIATIVO       boolean not null comment 'Campo que informa se o carro esta ativo ou não. Campo obrigatório.',
primary key (VEIID)
);
```

Apêndice C - Classe de comunicação com a Leitora RFID

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package condominium.applet;

import condominium.dao.VeiculoDAO;
import condominium.util.Conexao;
import java.io.IOException;
import java.net.InetAddress;
import java.net.MalformedURLException;
import java.net.UnknownHostException;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import java.util.TreeMap;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JFrame;
import javax.swing.table.DefaultTableModel;
import reader.Reader;

public class Rfid extends javax.swing.JApplet {

    // Declara as variáveis
    private javax.swing.JButton iniciar_btn;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JLabel nomePrograma_txt;
    private javax.swing.JTable veiculos_tab;
    private List lista;
    public DefaultTableModel dtm;
    public static Rfid instance = null;
    SimpleDateFormat dateFormat = new SimpleDateFormat("dd/MM/yyyy - HH:mm:ss");
    TreeMap<String, String> map = new TreeMap<String, String>();
    Conexao conexao = new Conexao();
    // Fim das declarações

    /** Inicia os componentes de RFID() */
    public Rfid() throws UnknownHostException {
        initComponents();
    }

    //Inicia as variáveis e a tela
    private void initComponents() throws UnknownHostException {
        InetAddress addr = InetAddress.getLocalHost();
        String ip = addr.getHostByName();
        System.out.println(ip);
        instance = this;
        jScrollPane1 = new javax.swing.JScrollPane();
        veiculos_tab = new javax.swing.JTable() {

            public boolean isCellEditable(int rowIndex, int colIndex) {
                return false; // Disallow the editing of any cell
            }
        };
        iniciar_btn = new javax.swing.JButton();
        nomePrograma_txt = new javax.swing.JLabel();
    }
}
```



```

private void iniciar_btnActionPerformed(java.awt.event.ActionEvent evt) throws SQLException,
UnknownHostException {
    Reader read = new Reader();
    Thread tread = new Thread(read);
    tread.start();
    iniciar_btn.setEnabled(false);
}

//Função para ao selecionar o item da tabela abrir o popup com os detalhes do veículo
private void veiculos_tabMouseClicked(java.awt.event.MouseEvent evt) throws MalformedURLException {
if (RfidDet.instance == null) {
    try {
        RfidDet.instance = new RfidDet(conexao, (String)
veiculos_tab.getValueAt(veiculos_tab.rowAtPoint(evt.getPoint()), 0), map.get((String)
veiculos_tab.getValueAt(veiculos_tab.rowAtPoint(evt.getPoint()), 0)));
    } catch (IOException ex) {
        Logger.getLogger(Rfid.class.getName()).log(Level.SEVERE, null, ex);
    }
    RfidDet.instance.setVisible(true);
} else {
    RfidDet.instance.dispose();
    RfidDet.instance.setVisible(false);
    try {
        RfidDet.instance = new RfidDet(conexao, (String)
veiculos_tab.getValueAt(veiculos_tab.rowAtPoint(evt.getPoint()), 0), map.get((String)
veiculos_tab.getValueAt(veiculos_tab.rowAtPoint(evt.getPoint()), 0)));
    } catch (IOException ex) {
        Logger.getLogger(Rfid.class.getName()).log(Level.SEVERE, null, ex);
    }
    RfidDet.instance.setVisible(true);
    RfidDet.instance.setExtendedState(JFrame.NORMAL);
}

    System.out.println(veiculos_tab.getValueAt(veiculos_tab.rowAtPoint(evt.getPoint()), 0));
}

//Método para listar os dados do veículo
public ArrayList<VeiculoDAO> listarVeiculo(String idVei, String idAntena) {
    ArrayList<VeiculoDAO> retorno = new ArrayList();
    ResultSet rs = null;
    PreparedStatement ps = null;
    try {
        String query = "select modNome,veicor,veiplaca,veiId from tb_veiculo, tb_modelo where veiId like ? and
modid= veiFk_modId";
        ps = (PreparedStatement) conexao.getConnection().prepareStatement(query);
        ps.setString(1, idVei);
        rs = ps.executeQuery();
        while (rs.next()) {
            VeiculoDAO vei = new VeiculoDAO();
            vei.setPlaca(rs.getString("veiplaca"));
            vei.setCor(rs.getString("veicor"));
            vei.setModeloNome(rs.getString("modNome"));
            vei.setVeiID(rs.getString("veiId"));
            retorno.add(vei);
            map.put(rs.getString("veiplaca"), idAntena);
            dtm.insertRow(0, new Object[] {vei.getPlaca(), vei.getModeloNome(), vei.getCor(), dateFormat.format(new
Date())});
        }
    } catch (Exception ex) {
        System.out.println("Erro: não foi possível procurar os registros, banco de dados temporariamente fora do
ar <br>Código do erro: " + ex.getMessage());
    } finally {
        try {
            if (rs != null) {

```

```

        rs.close();

    }
    if (ps != null) {
        ps.close();
    }
    } catch (Exception ex) {
System.out.println("Erro: não foi possível procurar os registros, servidor temporariamente fora do ar <br>Código
do erro: " + ex.getMessage());
    }
    }
    return retorno;
}

//Método para limpar os dados da tabela
public void limparTabela() {
    for (int i = 0; i < dtm.getRowCount(); i++) {
        dtm.removeRow(i);
    }
}

//Método principal para executar o applet
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {
            try {
                new Rfid().setVisible(true);
            } catch (UnknownHostException ex) {
                Logger.getLogger(Rfid.class.getName()).log(Level.SEVERE, null, ex);
            }
        }
    });
}
}

```

Apêndice D - Classe de leitura e gravação das Tags RFID

```
package condominium.applet2;

import condominium.util.Criptografia;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.Socket;

public class RfidGravar extends javax.swing.JFrame {
//Declara variáveis
private javax.swing.JButton gravar_btn;
private javax.swing.JTextArea gravar_txt;
private javax.swing.JLabel jLabel1;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JComboBox power_combo;
    private javax.swing.JLabel power_txt;
private javax.swing.JComboBox funcao_combo;
private javax.swing.JLabel funcao_txt;
    PrintWriter readerOut;
    BufferedReader readerIn;
    /** Cria construtor RfidGravar() */
    public RfidGravar() {
        initComponents();
    }

    //Inicia variáveis e a tela
private void initComponents() {

        gravar_btn = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        gravar_txt = new javax.swing.JTextArea();
        jLabel1 = new javax.swing.JLabel();
        funcao_combo = new javax.swing.JComboBox();
        power_combo = new javax.swing.JComboBox();
        power_txt = new javax.swing.JLabel();
        funcao_txt = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        gravar_btn.setLabel("Ler");
        gravar_btn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                gravar_btnActionPerformed(evt);
            }
        });

        gravar_txt.setColumns(20);
        gravar_txt.setRows(5);
        jScrollPane1.setViewportView(gravar_txt);

        jLabel1.setText("CONDOMINIUM");

        funcao_combo.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "Ler", "Gravar",
"Power" }));
        funcao_combo.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                funcao_comboActionPerformed(evt);
            }
        });
    }
}
```

```

power_combo.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "500", "1000", "2000",
"3000" }));

power_txt.setText("Power:");

power_txt.setVisible(false);
power_combo.setVisible(false);

funcao_txt.setText("Função:");

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addGroup(layout.createSequentialGroup()
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                                            .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 380,
Short.MAX_VALUE))
                                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()
                                            .addContainerGap(325, Short.MAX_VALUE)
                                            .addComponent(gravar_btn))
                                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()
                                            .addGap(158, 158, 158)
                                            .addComponent(jLabel1)))
                                    .addGroup(layout.createSequentialGroup()
                                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                            .addContainerGap()
                                            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                                                .addGroup(layout.createSequentialGroup()
                                                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                                        .addComponent(funcao_txt)
                                                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                                        .addComponent(funcao_combo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                                                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                                        .addComponent(power_txt)
                                                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                                                        .addComponent(power_combo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))))))
                                                .addContainerGap())
                                            .addComponent(funcao_combo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                                            .addComponent(funcao_txt))
                                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 21,
Short.MAX_VALUE)
                                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                                            .addComponent(funcao_combo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                                            .addComponent(funcao_txt))
                                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                                            .addComponent(power_txt)
                                            .addComponent(power_combo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                                        .addGap(18, 18, 18)
                                        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 138,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                        .addGap(18, 18, 18)

```

```

        .addComponent(gravar_btn)
        .addContainerGap()
    );

    pack();
}
//Função para gravar tag
private void gravar_btnActionPerformed(java.awt.event.ActionEvent evt) {
    gravar_txt.setText("");
    acessarLeitora();
}
//Função ao selecionar combobox
private void funcao_comboActionPerformed(java.awt.event.ActionEvent evt) {
if(funcao_combo.getSelectedItem()=="Ler"){
    gravar_btn.setLabel("Ler");
    power_txt.setVisible(false);
    power_combo.setVisible(false);
    gravar_txt.setVisible(true);
}
if(funcao_combo.getSelectedItem()=="Gravar"){
    gravar_btn.setLabel("Gravar");
    power_txt.setVisible(false);
    power_combo.setVisible(false);
}
if(funcao_combo.getSelectedItem()=="Power"){
    gravar_btn.setLabel("Power");
    power_txt.setVisible(true);
    power_combo.setVisible(true);
}
    System.out.println(funcao_combo.getSelectedItem());
    gravar_txt.setText("");
}
//Método de acesso a leitora
public void acessarLeitora() {
    try {
        Criptografia cript = new Criptografia();
        Socket socket = new Socket("192.168.0.10", 8080);
        readerOut = new PrintWriter(socket.getOutputStream(), true);
        readerIn = new BufferedReader(new InputStreamReader(socket.getInputStream()));
        execLeitora("RESET;");
        if(funcao_combo.getSelectedItem()=="Gravar"){
            execLeitora("UPDATE tag_id SET id = 0x"+Long.toHexString(cript.gerarIDUnico())+" where protocol_id = 12
            and antenna_id = 1;");
        }
        if(funcao_combo.getSelectedItem()=="Ler"){
            execLeitora("SELECT id,protocol_id, antenna_id FROM tag_id where protocol_id ='GEN2' and
            antenna_id = 1;");
        }
        if(funcao_combo.getSelectedItem()=="Power"){
            execLeitora("UPDATE saved_settings SET tx_power="+power_combo.getSelectedItem()+";");
        }
    } catch (Exception ex) {
        System.out.println(ex.getMessage());
        gravar_txt.setText(ex.getMessage());
    }
}

//Método para executar os comandos da leitora
public boolean execLeitora(String cmd) {
    String data = "";
    String readerEpc = "";
    readerOut.println(cmd);

    while (true) {

```

```

    try {
        data = readerIn.readLine();
    } catch (Exception e) {
        System.out.println(e.getMessage());
        gravar_txt.setText(e.getMessage());
    }
    if(data.length()>0){
        System.out.println("data.length() = " + data.length());
        System.out.println("data = " + data);
        gravar_txt.setText(data);
    }

    if (data.length() == 0) {
return false;
    }
}
}

//Método principal para executar o applet
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new RfidGravar().setVisible(true);
        }
    });
}
}

```

Apêndice E - Classe para executar a classe cancela

```
/*  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */
```

```
package Exemplo2;
```

```
public class Main {
```

```
/**
```

```
 * @param args the command line arguments
```

```
 */
```

```
public static void main(String[] args) {
```

```
 // TODO code application logic here
```

```
 try {
```

```
 (new TwoWaySerialComm()).connect("COM5");
```

```
 } catch (Exception e) {
```

```
 e.printStackTrace();
```

```
 }
```

```
 }
```

```
 }
```

Apêndice F - Classe comunicação cancela

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package Exemplo2;

import gnu.io.CommPort;
import gnu.io.CommPortIdentifier;
import gnu.io.SerialPort;
import java.io.FileDescriptor;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;

public class TwoWaySerialComm
{
    public TwoWaySerialComm()
    {
        super();
    }

    void connect ( String portName ) throws Exception
    {
        CommPortIdentifier portIdentifier = CommPortIdentifier.getPortIdentifier(portName);
        if ( portIdentifier.isCurrentlyOwned() )
        {
            System.out.println("Erro: A porta esta em uso");
        }
        else
        {
            CommPort commPort = portIdentifier.open(this.getClass().getName(),2000);

            if ( commPort instanceof SerialPort )
            {
                SerialPort serialPort = (SerialPort) commPort;
                serialPort.setSerialPortParams(2400,SerialPort.DATABITS_8,SerialPort.STOPBITS_1,Serial
                Port.PARITY_NONE);

                InputStream in = serialPort.getInputStream();
                OutputStream out = serialPort.getOutputStream();

                (new Thread(new SerialReader(in))).start();
                (new Thread(new SerialWriter(out))).start();

            }
            else
            {
                System.out.println("Erro: Somente as portas seriais são tratadas neste sistema.");
            }
        }
    }
}
```

```

    }

    /** */
    public static class SerialReader implements Runnable
    {
        InputStream in;

        public SerialReader ( InputStream in )
        {
            this.in = in;
        }

        public void run ()
        {
            byte[] buffer = new byte[1024];
            int len = -1;
            try
            {
                while ( ( len = this.in.read(buffer)) > -1 )
                {
                    System.out.print(new String(buffer,0,len));
                }
            }
            catch ( IOException e )
            {
                e.printStackTrace();
            }
        }

        /** */
        public static class SerialWriter implements Runnable
        {
            OutputStream out;

            public SerialWriter ( OutputStream out )
            {
                this.out = out;
            }

            public void run ()
            {
                try
                {
                    int c = 0;
                    while ( ( c = System.in.read()) > -1 )
                    {
                        this.out.write(c);
                    }
                }
                catch ( IOException e )
                {
                    e.printStackTrace();
                }
            }
        }
    }

```

```
    }  
  }  
}  
  
public static void main ( String[] args )  
{  
try  
  {  
    (new TwoWaySerialComm()).connect("COM4");  
  }  
catch ( Exception e )  
  {  
    // TODO Auto-generated catch block  
    e.printStackTrace();  
  }  
}  
}
```