

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ENGENHARIA INDUSTRIAL ELÉTRICA
ÊNFASE EM ELETRÔNICA/TELECOMUNICAÇÕES

EDUARDO JORDÃO SANTOS DE SÁ PALMEIRA
LUIZ RAFAEL DE ALMEIDA FERNANDES

**CONTROLE DE ACESSO PARA ESTÁDIOS DE FUTEBOL
UTILIZANDO TECNOLOGIA NFC(NEAR FIELD COMMUNICATION)**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2013

EDUARDO JORDÃO SANTOS DE SÁ PALMEIRA
LUIZ RAFAEL DE ALMEIDA FERNANDES

**CONTROLE DE ACESSO PARA ESTÁDIOS DE FUTEBOL
UTILIZANDO TECNOLOGIA NFC(NEAR FIELD COMMUNICATION)**

Projeto de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia Industrial Elétrica com Ênfase em Eletrônica e Telecomunicações do Departamento Acadêmico de Eletrônica – DAELN – da Universidade Tecnológica Federal do Paraná – UTFPR.

Professor orientador: Rubens Alexandre Faria

CURITIBA
2013

AGRADECIMENTOS

Agradecemos em primeiro lugar as nossas famílias, pela compreensão e suporte para realização desse projeto.

Ao orientador Rubens Faria, que tornou esse projeto possível.

A todos que direta ou indiretamente nos ajudaram, nos tornando mais fortes.

RESUMO

SANTOS DE SÁ PALMEIRA, Eduardo Jordão; DE ALMEIDA FERNANDES, Luiz Rafael. **Protótipo de comunicação NFC para controle de acesso**, 2013. Trabalho de Conclusão de Curso (Graduação) – Curso de Engenharia Industrial Elétrica: Ênfase em Eletrônica e Telecomunicações, UTFPR, Curitiba.

O objetivo principal desse projeto é a utilização da tecnologia *NFC* em uma implementação prática. Para isso foi escolhido um sistema completo de controle de acesso que pode ser empregado em estabelecimentos como estádios de futebol. A proposta deste trabalho é apresentar uma solução completa de controle de usuários que acessam um estabelecimento de forma prática e segura. Para tal será necessário um cadastro prévio de usuários e essas informações serão armazenadas em um banco de dados. Sendo assim o sistema possui três pilares: um banco de dados, um sistema que gerencia o banco de dados e realiza a interface com as catracas, e ainda as catracas. A comunicação da catraca com esse sistema gerenciador é feita através de rede *Ethernet*, utilizando o protocolo TCP/IP. A aplicação baseia-se na linguagem *C#*, o *firmware* na linguagem *C* e todo o banco de dados na linguagem *SQL*.

Palavras-chave: NFC. Automação de estádios. Wireless.

ABSTRACT

SANTOS DE SÁ PALMEIRA, Eduardo Jordão; DE ALMEIDA FERNANDES, Luiz Rafael. **NFC Communication Prototype for Access control**, 2013. Trabalho de Conclusão de Curso (Graduação) – Industrial Electrical Engineering Course: Emphasis in Electronics and Telecommunications, UTFPR, Curitiba.

The main objective of this project is the usage of the *NFC* technology in an application. To do this, it was chosen a complete system of access control that can be employed in establishments such as soccer stadiums. The purpose of this paper is to present a complete solution for controlling the users' access on an establishment in a practical and safe way. This will require a previous registration of the users and such information must be stored in a database. Thus the system has three pillars: a database, a system that manages the database and interfaces to the turnstiles, and even the turnstiles. The communication of the turnstiles with this management system is via Ethernet network, using TCP/IP protocol. The application is based on C# language, the firmware in C, and the entire database in language SQL.

Keywords: NFC. Stadium automation. Wireless.

LISTA DE FIGURAS

Figura 1 – Exemplos da compatibilidade da tecnologia NFC.	17
Figura 2 – Etiqueta RFID passiva.....	18
Figura 3 – NFC integrado via MicroSD.....	20
Figura 4 – Pagamento via NFC.	21
Figura 5 – Diagrama de estrutura de árvore Cliente – Conta Corrente.	23
Figura 6 – Diagrama de modelo em rede Cliente – Conta Corrente.	23
Figura 7 – Tabela do modelo relacional Cliente – Conta Corrente.....	24
Figura 8– Tabela Diagrama UML Cliente – Conta Corrente.....	25
Figura 9– Imagem Ilustrativa da controladora WKC IV.	29
Figura 10– Software, fornecido pela Wolpac, para controle do dispositivo e do banco de dados.....	30
Figura 11– Representação ilustrativa do sistema.....	31
Figura 12– Fluxograma do sistema.	32
Figura 13– Arduino Duemilanove.	33
Figura 14 – Esquemático Arduino Duemilanove.	35
Figura 15– Leitor ACR122S.	36
Figura 16– ACR122STool.	37
Figura 17 – Protocolo de comunicação ACR122S.	37
Figura 18 – Fluxo do protocolo de comunicação ACR122S.....	38
Figura 19 – Leitura de dados da comunicação serial do leitor ACR122S	39
Figura 20 – Descrição de cabeçalho de comando de troca de dados.....	40
Figura 21 – Descrição de cabeçalho de resposta de troca de dados.....	41
Figura 22 – Shield Ethernet DFRobot	42
Figura 23 – Display LCD 16x2.....	43
Figura 24 – Foto do teclado utilizado	44
Figura 25 – Catraca Wolstar.....	45
Figura 26 – Funcionamento da catraca.....	47
Figura 27 – Tabela de informações de clientes.....	49
Figura 28 – Tabela de informações de clientes.....	50
Figura 29 – Trecho do código – Cadastrando Clientes	50
Figura 30 – Tela Principal da Plataforma	51

Figura 31 – Trecho do código – Construtora da classe Clientes	52
Figura 32 – Janela para cadastro de cartões	53
Figura 33 – Janela para excluir cartões	55
Figura 34 – Trecho código – Botão Excluir Cartão	55
Figura 35 – Janela para editar cartões.....	56
Figura 36 – Janela para alterar acesso de clientes	57
Figura 37 – Trecho código – Botão Reset Server	58
Figura 38 – Janela para monitorar os dados recebidos	59
Figura 39 – Primeiro protótipo	60
Figura 40 – Software Hercules	61
Figura 41 - Modelo de Negócio Canvas da empresa RDAccess.....	76

LISTA DE TABELAS

Tabela 1– Características Arduino Duemilanove	34
Tabela 2 – Quadro com o comando de leitura de um cartão no ACR122S	39
Tabela 3 - Descrição do cabeçalho de comando no ACR122S	40
Tabela 4 – Quadro com resposta ao comando de leitura de um cartão no ACR122S	41
Tabela 5 - Descrição do cabeçalho de resposta no ACR122S	41
Tabela 6– Custos do sistema	63
Tabela 7– Equipamentos e Mobiliário da Sede da Empresa	78
Tabela 8– Gastos Iniciais	78
Tabela 9 – Gastos Mensais	79
Tabela 10– Fluxo de Caixa Esperado	80

LISTA DE SIGLAS

BD - Banco de dados

bps – bits por segundo

CLR - Common Language Runtime (Ambiente de Execução Independente de

IP – Internet Protocol (Protocolo Internet)

LCD – Liquid crystal display (Display de cristal líquido)

LED - Light Emitting Diode (Diodo Emissor de Luz)

Linguagem)

NFC – Near Field Communication (Comunicação em área próxima)

RFID - Radio-Frequency Identification (Identificação por radiofrequência)

RG – Registro geral

SGBD - Sistemas Gerenciadores de Banco de Dados

SPI - Serial Peripheral Interface (Interface Periférica Serial)

SQL - Structured Query Language (Linguagem de Consulta Estruturada)

TCP - Transmission Control Protocol (Protocolo de controle de transmissão)

UART - Universal Asynchronous Receiver/Transmitter (Receptor/Transmissor
Assíncrono Universal)

USB - Universal Serial Bus (Barramento Serial Universal)

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVO	13
1.2	JUSTIFICATIVA	14
1.3	METODOLOGIA	14
2	FUNDAMENTAÇÃO TEÓRICA	16
2.1	NEAR FIELD COMMUNICATION (NFC)	16
2.1.1	Modos de operação	18
2.1.2	Características da comunicação	18
2.1.3	Segurança	19
2.1.4	NFC incorporado a dispositivos móveis	19
2.2	BANCO DE DADOS	21
2.2.1	Sistemas Gerenciadores de Bancos de Dados	22
2.2.2	Linguagem SQL	25
2.3	PLATAFORMA .NET	26
2.3.1	Linguagem C#	27
3	DESENVOLVIMENTO DO PROJETO	29
3.1	FUNCIONAMENTO BÁSICO	31
3.2	COMPONENTES DO SISTEMA	33
3.2.1	Catraca	33
3.2.2	Servidor com banco de dados e interface com o usuário	48
4	RESULTADOS OBTIDOS	60
4.1	TESTES E VERIFICAÇÕES	60
4.2	ADMINISTRAÇÃO DO SISTEMA E FUNCIONALIDADES	62
4.3	CONFIABILIDADE	62
4.4	CUSTO	63
5	PLANO DE NEGÓCIOS	65
5.1	DEFINIÇÃO DO NEGÓCIO	65
5.1.1	Visão	65
5.1.2	Missão	65
5.1.3	Valores	65
5.1.4	Descrição do Negócio	65
5.2	OBJETIVOS	66
5.2.1	Objetivos Principais	66
5.2.2	Objetivos Intermediários	66
5.3	DEFINIÇÃO DO PRODUTO	67
5.3.1	Descrição	67
5.3.2	Análise Comparativa	67
5.3.3	Tecnologia	68
5.3.4	Produtos e Serviços Futuros	69
5.4	DEFINIÇÃO DO MERCADO	69
5.4.1	Segmentação de mercado	69
5.4.2	Segmento Alvo	69
5.4.3	Necessidade	69
5.4.4	Tamanho	70
5.4.5	Crescimento	70
5.4.6	Players	70
5.4.7	Modelo de distribuição	71
5.4.8	Modelo de Competitividade	71

5.5 OFERTA E A PROPOSTA DE VALOR.....	71
5.6 ESTRATÉGIA E IMPLEMENTAÇÃO	72
5.6.1 Diferenciais Competitivos e Proposta de Valor	72
5.6.2 Estratégia de Marketing	73
5.7 ESTRUTURA ORGANIZACIONAL	74
5.7.1 Estrutura Legal.....	74
5.7.2 Infraestrutura.....	74
5.8 MODELO DE NEGÓCIO CANVAS	76
5.9 PLANO FINANCEIRO	77
5.9.1 Contabilização	77
5.9.2 Custo.....	78
5.9.3 Fluxo de Caixa Esperado.....	79
6 CONCLUSÃO	81
6.1 CONSIDERAÇÕES SOBRE O DESENVOLVIMENTO DO PROJETO.....	81
6.2 OBJETIVOS ALCANÇADOS	82
REFERÊNCIAS.....	83

1 INTRODUÇÃO

Desenvolvida em uma parceria entre os gigantes Sony e Phillips, a chamada *Near Field Communication* (NFC, comunicação em área próxima) permite a comunicação, leia-se troca de dados, entre dois dispositivos eletrônicos compatíveis que estejam a poucos centímetros de distância ou encostados, de forma extremamente rápida e segura. No caso, primeiramente haveria a identificação de um dispositivo pelo outro e em um segundo momento ocorreria a transmissão dos dados, efetivamente (Peron, 2009).

Essa tecnologia, que vem surgindo como uma tendência para os novos dispositivos móveis permite diversas aplicações, entre elas: controle de acesso, coleta e troca de informações, cartões fidelidade e cupons, pagamentos, transporte. No projeto desenvolvido será mostrada a aplicação de controle de acesso.

Não há dúvidas que controle de acesso seja imprescindível em estabelecimentos com alta frequência de público ou necessidade de segurança, como estádios, teatros, casas de show, hotéis, cofres, etc.

No dia 29 de março de 2013, a Polícia Militar usou bomba de efeito moral e gás lacrimogêneo após uma confusão entre torcedores do Esporte Clube Bahia, na fila para venda de ingressos na Arena Fonte Nova, estádio sede da Copa das Confederações e Copa do Mundo em Salvador (Marques, 2013). Diante desse cenário, se faz necessário unir alta segurança e praticidade se tratando de acesso. Sendo assim NFC é capaz de atender esses objetivos e fazer com que situações como essa não se repitam.

Comunicação NFC, difere de outras comunicações sem fio como RFID ou *Bluetooth*, pela distância de funcionamento, já que o NFC só funciona a uma distancia máxima de 20 cm (Peron, 2009). Isso traz uma enorme segurança ao usuário, visto que fica bem complicado de um terceiro capturar os dados transmitidos durante a operação.

Quem sabe como será aplicada essa comunicação no futuro? Afinal, a tecnologia não parece encontrar obstáculos para evoluir. Pensando assim, talvez as chaves comuns pudessem ser substituídas: no lugar de trincos e fechaduras, existiriam leitores de chaves eletrônicas. Também se pode imaginar em carteiras de

identidade eletrônicas, onde apresentaria uma espécie de cartão quando o segurança pedisse seu documento para entrar em algum bar (Peron, 2009).

A tendência é que ao passar dos anos o NFC esteja presente em todos os dispositivos móveis e seja possível adquirir acesso a estabelecimentos via rede Internet. Tendo em vista que tal fato ainda não é possível, aplicação aqui desenvolvida será um controle de acesso, baseado em banco de dados, com interface de usuário, que se faz necessário cadastro prévio dos usuários. Uma vez cadastrado, considerando que o cartão ou “tag” NFC será propriedade de cada cliente, para os próximos acessos serão necessário apenas aquisição da liberação para entrada, que poderá ser feita via rede, bilheteria, ou seja, a critério do estabelecimento. Ainda é possível conceder créditos de acesso a cada cliente, o que poderia ser aplicado, por exemplo, em um programa sócio torcedor de estádio de futebol. Visando assim, reduzir as grandes filas em estabelecimentos, e acabando com a carteira sócio torcedor.

De fato, essa tecnologia ainda é um embrião nascendo, mesmo após dois anos de vendas de aparelhos com o chamado elemento seguro de transmissão. Mas o que fica claro é que todo mundo está se mobilizando para entender esse mercado, e o impacto não vai demorar a chegar a terras brasileiras (Zanni, 2013).

1.1 OBJETIVO

A proposta consiste no estudo e desenvolvimento de uma aplicação para a tecnologia NFC. Para isso será utilizado o mecanismo de atuação de uma catraca, que permitirá apenas acesso após verificação. Assim será adquirida a catraca Wolstar II da empresa Wolpac. Toda interface com o usuário e o controle de acesso será desenvolvida nesse projeto.

O projeto também contempla o desenvolvimento de um aplicativo *windows*, que manipulará o banco de dados com informações referente à liberação ou não da catraca, e também a comunicação com demais catracas.

O objetivo, então, consiste no desenvolvimento de um sistema de controle de acesso, utilizando a tecnologia NFC.

1.2 JUSTIFICATIVA

A tecnologia surgiu a partir do RFID (*Radio Frequency Identification*). Trata-se de uma tecnologia recente e promissora, consolidada no mercado europeu e asiático.

Além disso, não existe um sistema completo de controle de acesso, com as funcionalidades que serão, neste trabalho, apresentadas. Ainda, este sistema contribui para segurança e praticidade dos usuários.

Este projeto foi planejado, também com o intuito de desenvolver o conhecimento técnico dos envolvidos. Como dito, essa tecnologia sendo recente, possibilitou o acesso de um grande campo de pesquisa.

1.3 METODOLOGIA

As etapas do projeto, visando atingir os objetivos com êxito, serão:

- Estudo da tecnologia NFC;
- Pesquisa de mercado para escolha dos equipamentos que serão utilizados;
- Estudo da catraca Wolstar II adquirida junto a Wolpac;
- Estudo da placa de controle WKC IV;
- Testes com a placa de controle WKC IV mais catraca Wolstar II;
- Estudo do leitor NFC ACR122S previamente adquirido;
- Estudo das "Tags" compatíveis com o ACR122S;
- Testes com o leitor ACR122S mais Tags, utilizando um microcomputador;
- Testes de integração do leitor NFC ACR122S na placa de controle WKC IV presente na catraca Wolstar II;
- Escolha de um kit de desenvolvimento para realização do protótipo;
- Testes de integração do Leitor NFC ACR122S em kit de desenvolvimento;
- Escolha de periféricos para interface entre usuário e catraca;
- Desenvolvimento de firmware para kit de desenvolvimento;

Desenvolvimento de aplicativo, com interface para o usuário, para controle da integração;

Desenvolvimento de aplicativo para administrar rede de catracas, com banco de dados e interface para usuários;

Teste de validação do protótipo.

2 FUNDAMENTAÇÃO TEÓRICA

A estrutura básica de um sistema de controle de acesso deve possuir as seguintes fases: coleta de dados do usuário, tratamento dos mesmos, acesso ao banco de dados e tomada de decisão.

Existem várias formas para coleta de dados de usuários já empregadas em sistemas de controle de acesso, entre elas biometria, cartão RFID, senha e códigos de barras. Nesse projeto, será utilizado o NFC, tecnologia recente no cenário brasileiro.

Será utilizada uma catraca real, e a mesma possuirá uma placa controladora que receberá os dados do usuário e comunicará via rede TCP/IP com uma interface manipuladora do banco de dados, desenvolvida em linguagem C#.

Sendo assim, o usuário deverá primeiramente comprar o acesso, e caso não possua cadastro, já efetuar seu cadastro no banco de dados. Após acesso liberado pela bilheteria, deverá solicitar seu acesso junto à catraca, então a placa controladora irá enviar os dados para plataforma manipuladora desenvolvida em C#, a mesma irá acessar o banco de dados e responder à placa manipuladora da catraca a decisão a ser tomada quanto ao acesso.

2.1 NEAR FIELD COMMUNICATION (NFC)

NFC é a sigla para *Near Field Communication* – Comunicação por campo próximo. NFC é uma tecnologia de radiofrequência, diferencia-se das demais pela distância de operação, em torno de 0 a 20 cm entre os dispositivos.

A ideia do NFC é tornar a comunicação entre os dispositivos mais simples: basta aproximá-los para que isso ocorra, eliminando a necessidade de procedimentos eventualmente complexos. Pelo fato da distância de operação ser curta, a comunicação é inerentemente segura no que diz respeito a tentativas de interceptação. Caso haja necessidade de coibir esse tipo de ação, as medidas usuais de segurança devem ser no protocolo de rede e/ou aplicação (ATOJI, 2010).

Desenvolvido pela parceria das empresas Philips Semiconductors e Sony Corporation, NFC opera a 13,56MHz, com taxas de transferência de 106 a 848kbps.

Considera-se o NFC a segunda geração da tecnologia RFID e espera-se que vários dispositivos incorporem essa tecnologia. A mesma é compatível com o SmartCard “Felica™” desenvolvido pela Sony e utilizado em países como Japão, e também como “Mifare”, SmartCard desenvolvido pela Philips (LEVANDOVSKI, 2011). Na figura 1 é possível ver exemplos de compatibilidade da tecnologia NFC.

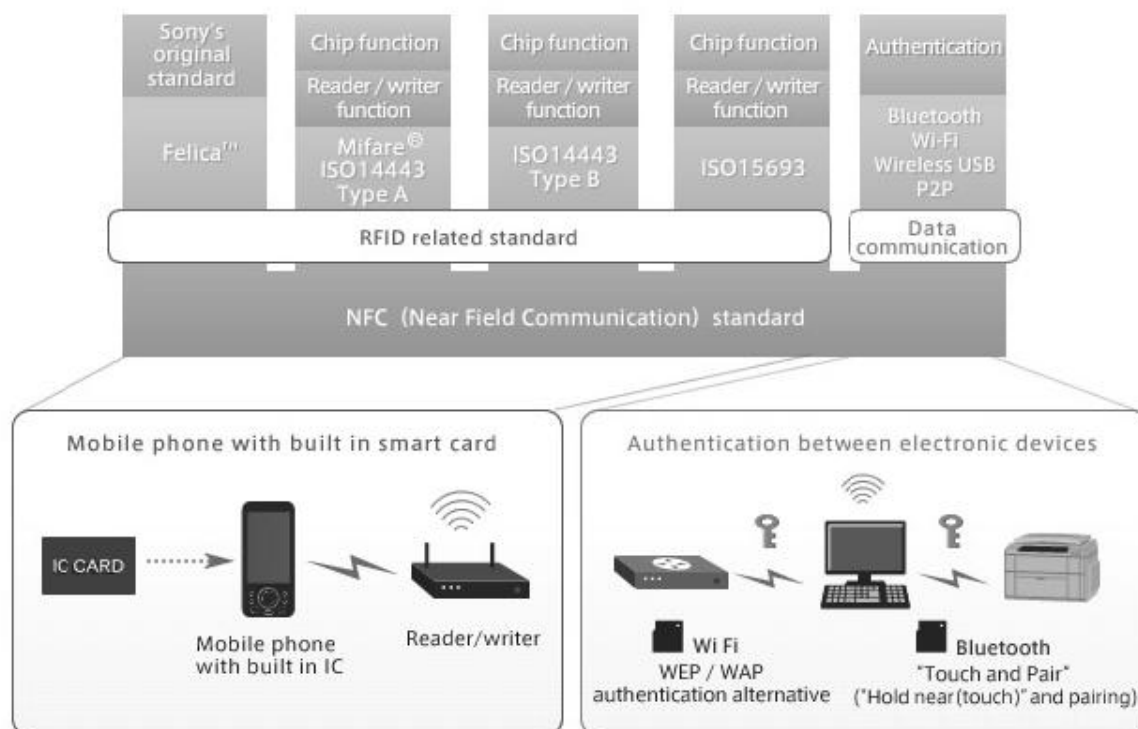


Figura 1 – Exemplos da compatibilidade da tecnologia NFC.

Fonte - (LEVANDOVSKI, 2011)

Para que dois dispositivos se comuniquem usando tecnologia NFC, um dispositivo deve ter um leitor/gravador NFC e o outro deve ter uma etiqueta NFC. A etiqueta é essencialmente um circuito integrado que contém dados, conectado a uma antena, que pode ser lido e escrito pelo dispositivo (LEVANDOVSKI, 2011).

Dito isso, a tecnologia NFC pode ser utilizada para vários propósitos, como transmissão de pequena quantidade de dados, incluindo transações bancárias, cartões de visita, autenticação, ou o pode ser também iniciador de comunicação secundária, como *Bluetooth* ou *WiFi*.

2.1.1 Modos de operação

Existem dois modos de operação:

- Modo passivo: apenas um dispositivo gera sinal de radiofrequência para comunicação. Neste caso o campo eletromagnético gerado irá alimentar eletricamente o dispositivo passivo, viabilizando a comunicação. Assim funcionaria a integração NFC de aparelhos móveis com cartazes, anúncios e cartões, sendo o dispositivo móvel o ativo. Na figura 2 está ilustrado uma etiqueta passiva.

- Modo Ativo: Ambos os dispositivos são capazes de gerar sinais de radiofrequência para comunicação. Nesse caso, o campo magnético gerado, induz corrente elétrica no dispositivo passivo.



Figura 2 – Etiqueta RFID passiva.

Fonte - (ATOJI, 2010)

2.1.2 Características da comunicação

Trata-se de um protocolo *peer-to-peer*, ou seja, o número de dispositivos participantes da comunicação é limitado a dois. Utiliza uma única banda de radiofrequência, isso implica em apenas um dispositivo pode transmitir por vez, sendo, portanto uma comunicação *half-duplex*. É necessário que o dispositivo que

vá iniciar a transmissão verifique se não tem ninguém transmitindo dados no momento, para só então iniciar a transmissão. (ATOJI, 2010).

2.1.3 Segurança

Essa tecnologia tem como objetivo agregar maior agilidade e praticidade em transações, garantindo maior segurança. Para isso o NFC conta com sua maior característica: a pequena distância entre os dispositivos para comunicação.

Além dessa característica, está presente também o fato de o processo ocorrer em poucos segundos.

É necessário ainda, empregar processos de autenticação e criptografia. Um aparelho iniciador pode solicitar a autenticação do usuário antes de realizar a comunicação e transferência de informações. Vários tipos de criptografias podem ser empregados, sendo a utilizada determinada pela característica de cada aplicação.

2.1.4 NFC incorporado a dispositivos móveis

Para que o NFC possa vir a ser utilizado em massa, é necessário grande disponibilidade de equipamentos que possuam a tecnologia NFC integrada. Atualmente não existem muitos aparelhos disponíveis, mas estima-se que grande parte dos dispositivos móveis produzidos introduza essa tecnologia. Para os modelos atuais que não possuam essa característica, já existe uma solução de adaptação da tecnologia. Esta solução são antenas e adaptadores para o cartão SIM ou a utilização de um cartão MicroSD específico. Como é o caso do modelo *In2play*, desenvolvido pela empresa Device Fidelity (figura 3).



Figura 3 – NFC integrado via MicroSD.

Fonte - (LEVANDOVSKI, 2011)

Dentro desse contexto, a multinacional Samsung deu um passo a frente:

A Visa e a Samsung anunciam uma aliança global que combina a experiência Visa nos pagamentos com a liderança tecnológica da Samsung em dispositivos móveis. Uma aliança que tem um potencial de acelerar significativamente a disponibilização global de pagamentos em dispositivos móveis. Segundo o acordo, as instituições financeiras que estão a planear o lançamento de programas de pagamentos móveis poderão usar o Visa Mobile Provisioning Service para transferirem, de uma forma segura, as informações das contas para os dispositivos móveis Samsung com NFC. Para, além disso, a Samsung concordou em colocar a aplicação *Visa payWave* nos seus dispositivos móveis com tecnologia NFC. *Visa payWave* é uma aplicação de pagamentos móveis que permite aos consumidores fazerem pagamentos *contactless* usando os seus dispositivos móveis, ilustrado na figura 4 (SILVA, 2013).



Figura 4 – Pagamento via NFC.

Fonte – (SILVA, 2013)

Estima-se um alto crescimento nos próximos anos dessa tecnologia já presente no Brasil. Ao utilizar a mesma como ferramenta de *marketing*, acesso a informações, controle de acesso é mais simples. Quando trata-se de efetuar pagamentos com a mesma envolve uma operação mais complexa, uma vez que envolve vários *players*: bancos, redes de captura, empresas de cartão de crédito, estabelecimentos comerciais, operadoras de serviços móveis. Mas sim, essa tecnologia é uma tendência global.

2.2 BANCO DE DADOS

Entende-se como banco de dados (BD) uma coleção de informações logicamente coerente, em uma determinada ordem cuja interpretação é dada por uma determinada aplicação. São de vital importância para grupos, empresas e organizações, tornando os processos mais práticos e seguros.

2.2.1 Sistemas Gerenciadores de Bancos de Dados

Um Sistema de Bancos de Dados consiste em uma base de dados e um *software* que o manipula. Esse *software* que o manipula é o Sistema Gerenciador de Bancos de Dados (SGDB), que nada mais é que um *software* construído para facilitar as atividades de definição, construção e manipulação de bancos de dados.

Sem um sistema gerenciador do banco de dados o mesmo fica sujeito a operação manual, com iminência de falha por parte do operador. A fim de minimizar esses problemas o SGDB proporciona proteção a acessos não autorizados, acesso eficiente a dados para consulta e modificação, e controle de acessos simultâneos evitando que os dados sejam corrompidos.

Normalmente, um SGDB é estruturado em linguagem do tipo *query*, as quais são de alto nível, simplificando consideravelmente a fase de desenvolvimento de aplicativos capazes de manipular a base de dados.

2.2.1.1 Modelos de Dados

O primeiro Sistema Gerenciador de Banco de Dados (SGBD) comercial surgiu no final de 1960 com base nos primitivos sistemas de arquivos disponíveis na época, os quais não controlavam o acesso concorrente por vários usuários ou processos. Os SGBD's evoluíram desses sistemas de arquivos de armazenamento em disco, criando novas estruturas de dados com o objetivo de armazenar informações. Com o tempo, os SGBD's passaram a utilizar diferentes formas de representação, ou modelos de dados, para descrever a estrutura das informações contidas em seus bancos de dados (TAKAI, 2005).

Hoje, os modelos mais utilizados pelos sistemas gerenciadores são: modelo hierárquico, modelo em redes, modelo relacional e o modelo orientado a objetos.

- **Modelo Hierárquico:** Primeiro a ser reconhecido como modelo de base de dados. Ele conecta registros numa estrutura de dados em árvore ou hierarquias, os nós das hierarquias contêm ocorrência de registros, sendo cada registro uma coleção de atributos. O registro da árvore que precede a outros é o

registro-pai, os outros são chamados de registros-filhos, a navegação é do topo para baixo, e da esquerda para direita. O sistema mais conhecido no modelo hierárquico foi o *Information Management System* da IBM Corp (IMS). A figura 5 descreve o esquema de um banco de dados hierárquico, sendo que as caixas correspondem aos tipos de registros e as linhas às ligações entre os tipos de registros (TAKAI, 2005).

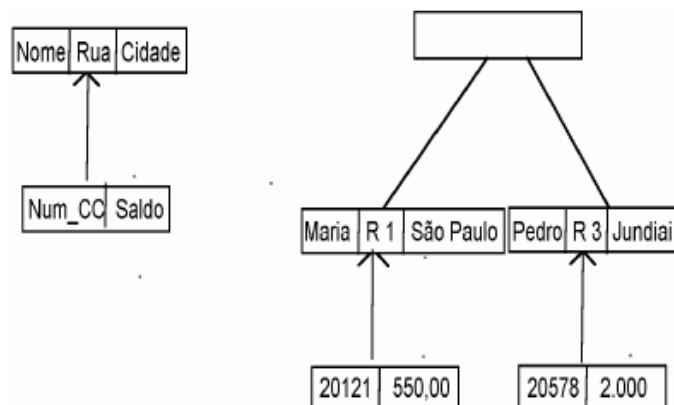


Figura 5 – Diagrama de estrutura de árvore Cliente – Conta Corrente.

Fonte – (TAKAI, 2005)

- Modelo em Redes:** Surgiu como uma extensão ao modelo hierárquico, eliminando a hierarquia e permitindo que um registro filho se conecte a vários registros pais, criando várias associações. Diferentemente do modelo em árvore, em que qualquer acesso passa pela raiz, o modelo em rede permite acesso a qualquer nó da rede sem passar pela raiz. O sistema comercial mais divulgado nesse modelo é CAIDMS da Computer Associates. A figura 6 ilustra esse modelo (TAKAI, 2005).

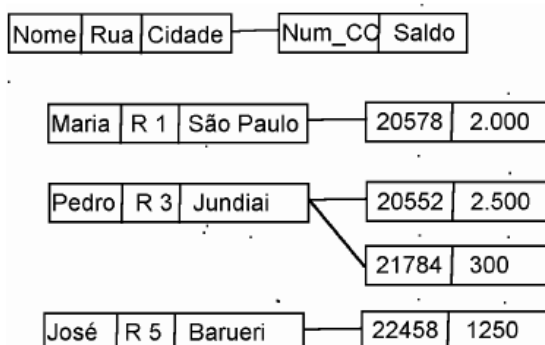


Figura 6 – Diagrama de modelo em rede Cliente – Conta Corrente.

Fonte – (TAKAI, 2005)

- **Modelo Relacional:** Sua definição é teórica e baseada na teoria de conjuntos. Seus objetivos eram melhorar a independência de dados nos sistemas gerenciadores de banco de dados e prover um conjunto de funções apoiadas em álgebra relacional para armazenamento e recuperação de dados.

A estrutura básica do modelo relacional é a relação (tabela), constituída por um ou mais atributos (campos) que traduzem o tipo de dados a armazenar. Cada linha do esquema é chamada de tupla (registro). O modelo relacional não tem caminhos predefinidos para acesso aos dados, como os modelos que o antecedem. Porém para trabalhar com essas tabelas algumas restrições devem ser impostas para evitar ações indesejáveis, como repetição de informação e perda de informação. Atualmente é o modelo mais utilizado, aparecendo em 80% dos bancos de dados existentes, a figura 7 ilustra um exemplo desse modelo (TAKAI, 2005).

Cod_Cliente	Nome	Rua	Cidade
1	Pedro	A	São Paulo
2	Maria	B	Jundiai

Num_CC	Saldo	Cod_Cliente	Num_CC
20121	1200	1	20121
21582	1320	2	21582
21352	652	2	21352

Figura 7 – Tabela do modelo relacional Cliente – Conta Corrente.

Fonte – (TAKAI, 2005)

- **Modelo Orientado a Objetos:** Os bancos de dados orientados a objeto começaram a se tornar comercialmente viáveis em meados de 1980. A motivação para seu surgimento está em função dos limites de armazenamento e representação semântica impostas no modelo relacional. Alguns exemplos são os sistemas de informações geográficas (SIG), os sistemas CAD e CAM, que são mais facilmente construídos usando tipos complexos de dados. A habilidade para criar os tipos de dados necessários é uma característica das linguagens de programação orientadas a objetos (TAKAI, 2005).

Tudo indica que o banco de dados orientado a objetos serão utilizados apenas em aplicações específicas, enquanto os sistemas relacionais continuarão a dominar o mercado. A figura 8 ilustra o modelo orientado a objetos.

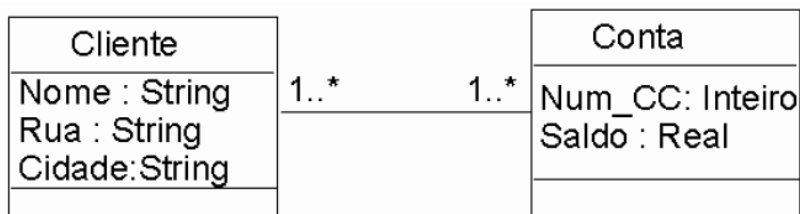


Figura 8– Tabela Diagrama UML Cliente – Conta Corrente.

Fonte – (TAKAI, 2005)

2.2.2 Linguagem SQL

Structured Query Language, ou Linguagem de Consulta Estruturada ou SQL, é uma linguagem de pesquisa declarativa, para banco de dados relacional.

Foi desenvolvida originalmente no início dos anos 70 nos laboratórios da IBM. Pertencia ao projeto System R, que tinha por objetivo demonstrar a viabilidade da implementação do modelo relacional proposto anteriormente.

Apesar de ser originalmente criada pela IBM, muitos desenvolvedores foram criando “dialetos” para ela. Essa expansão levou à necessidade de ser criado e adaptado um padrão para a linguagem. Assim em 1987 a linguagem SQL foi padronizada pela ANSI e ISSO e revisada nos anos de 1992,1999 e 2003.

Graças a sua simplicidade e facilidade de uso, a linguagem SQL é quase um padrão em bancos de dados. A diferença desta para outras linguagens é, principalmente, que o foco está apenas nos resultados, e não necessariamente na forma como estes foram obtidos. É uma linguagem altamente intuitiva, declarativa, ao contrário de outras procedurais (ELMASRI; NAVATHE, 2004, p.105).

As instruções da linguagem SQL são divididas basicamente em cinco subconjuntos:

- DML – Linguagem de Manipulação de Dados;
- DDL – Linguagem de Definição de Dados;
- DCL – Linguagem de Controle de Dados;
- DTL – Linguagem de Transação de Dados;

As instruções do grupo DML permitem realizar inclusões, consultas, alterações e exclusões de dados presentes em registros. Estas tarefas podem ser realizadas em vários registros de diversas tabelas simultaneamente. São comandos pertencentes a esse grupo: INSERT, SELECT, UPDATE e DELETE.

Já as instruções pertencentes ao grupo DDL (*Data Definition Language* – Linguagem de Definição de Dados) permite ao utilizador definir tabelas novas e elementos associados. São comandos básicos DDL: CREATE, DROP, ALTER, TRUNCATE, COMMENT E RENAME.

O próximo conjunto é o DCL (*Data Control Language*, Linguagem de Controle de Dados) que controla as autorizações e licenças de usuários limitando o acesso, leitura e manipulação de dados. Duas instruções-chaves da DCL: GRANT e REVOKE.

O quarto conjunto é o DTL, que gerencia as mudanças feitas pelo grupo DML cujo grupo permite que instruções sejam agrupadas em padrões lógicos. São instruções típicas: COMMIT, SAVEPOINT, ROLLBACK, SET TRANSACTION.

2.3 PLATAFORMA .NET

A plataforma *Microsoft* oferece suporte a aplicativos e soluções projetadas para os menores dispositivos, bem como para as maiores empresas. Os desenvolvedores de *software* que usam a plataforma *Microsoft* podem:

- Trabalhar com uma variedade de linguagens, padrões e ferramentas de programação.
- Criar aplicativos para o *Windows*, o *Office*, a *Web* e para os dispositivos móveis.

- Beneficiar-se dos recursos que a *Microsoft* oferece para ajudar os desenvolvedores de *software* a criar e projetar aplicativos de classe mundial. (MSDN,2013).

O objetivo da plataforma é de se tornar única para desenvolvimento e execução de sistemas e aplicações. Todas as aplicações concebidas nessa plataforma podem ser executadas em qualquer dispositivo que contenha um framework .NET.

Utiliza como base um ambiente chamado *Common Language Runtime* (CLR - Ambiente de Execução Independente de Linguagem) atuando com um conjunto de bibliotecas unificadas, ou *framework*. A CLR é capaz de suportar, atualmente, 33 linguagens diferentes de programação, que se relacionam como se fossem uma única linguagem (MSDN, 2013).

2.3.1 Linguagem C#

C# é uma linguagem elegante e de tipos protegidos, orientada a objeto e que permite aos desenvolvedores construir uma variedade de aplicações seguras e robustas, compatíveis com o .NET Framework. Pode-se usar o C# para criar aplicações tradicionais do Windows, Web services baseados em XML, componentes distribuídos, aplicativos cliente-servidor, aplicativos com banco de dados e mais. O Visual C# 2008 oferece um avançado editor de código, uma interface de usuário conveniente, depurador integrado e muitas outras ferramentas para facilitar o desenvolvimento de aplicativos baseados na versão 3.0 da linguagem C# e na versão 3.5 do .NET Framework (MSDN,2013).

A sintaxe do C# é altamente expressiva, mas ela também é simples e fácil de aprender. A sintaxe do C# será instantaneamente reconhecida por qualquer pessoa familiarizada com C, C++ ou Java. Os desenvolvedores que sabem qualquer uma dessas linguagens são geralmente capazes de começar a trabalhar de forma produtiva com C# dentro de um tempo muito curto. A sintaxe do C# simplifica muitas das complexidades do C++ e fornece recursos poderosos, como tipos de valor nulo, enumerações, delegações, expressões *lambda* e acesso direto a memória, que não são encontrados no Java (MSDN,2013).

Como uma linguagem orientada à objetos, o C# suporta os conceitos de encapsulamento, herança e polimorfismo. Todas as variáveis e métodos, incluindo o método principal (Main), o ponto de execução de uma aplicação, são encapsuladas em definições de classes. Uma classe derivada pode herdar diretamente somente de uma classe pai, mas pode herdar de qualquer quantidade de interfaces. Métodos da classe derivada que substituem métodos virtuais de uma classe pai exigem a utilização da palavra-chave `override` como forma de evitar a redefinição acidental (MSDN,2013).

O uso dos chamados *namespaces* é outra grande marca do C#. *Namespaces* são maneiras de organizar os diferentes tipos que ocorrem em uma aplicação C#. Seu conceito é um pouco semelhante a uma pasta do sistema de arquivos em um computador. Cada *namespace* contém uma série de classes, que corretamente instanciadas, podem ser utilizadas no desenvolvimento da aplicação (MSDN, 2013).

3 DESENVOLVIMENTO DO PROJETO

O projeto descrito consiste em uma aplicação para a tecnologia NFC, tal aplicação é uma solução de controle de acesso, em princípio idealizados para uso em estádios de futebol.

Primeiramente foi adquirido o leitor NFC ACR122S e a catraca Wolstar da empresa Wolpac. Essa catraca possui uma placa controladora modelo WKC IV com um *firmware* próprio e um *software* a ser instalado no servidor para configuração da catraca, gerenciamento e criação de banco de dados. O próprio fabricante do dispositivo Wolstar informou que a placa controladora possibilitaria configuração com vários leitores como biométrico, *mifare*, e possivelmente NFC, dado que a mesma possui portas seriais, USB, além de dispositivos para controle de potência disponíveis. Sendo assim o primeiro caminho tomado para realização do projeto, foi integrar o leitor ACR122S a WKCIV. Para isso foi realizado um longo período de estudo e análise das ferramentas, incluindo contato direto com representantes da Wolpac. Ao fim desse estudo, constatou-se a inviabilidade da utilização do NFC na controladora WKCIV, devido a mesma não possibilitar acesso completo a configuração da COM1 disponível na controladora da catraca, não podendo selecionar os caracteres a serem escritos, e também o número de *bytes* a ser lido por vez. Na figura 9 é possível ver a imagem ilustrativa da controladora WKC IV e na figura 10 a imagem da tela inicial do software fornecido pela Wolpac, para controle do dispositivo e do banco de dados.

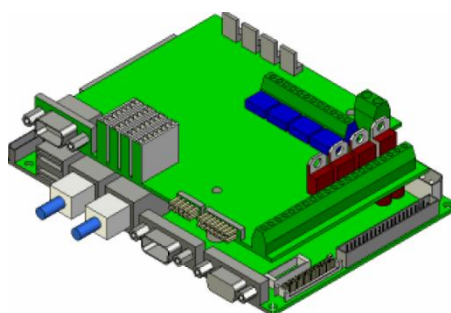


Figura 9– Imagem Ilustrativa da controladora WKC IV.

Fonte – (Adaptado de Departamento de Engenharia Wolpac,2011)



**Figura 10– Software, fornecido pela Wolpac, para controle do dispositivo e do banco de dados.
Fonte – Autoria própria.**

Sabendo do problema anteriormente descrito, foi reformulado o plano de ação do projeto. Decidiu-se então pelo desenvolvimento do mecanismo de atuação, mais precisamente uma catraca, selecionando então uma placa controladora, e a criação de um *software* para comunicação com o mesmo e controle do banco de dados. Sendo assim a figura 11 representa os principais componentes da solução empregada. Foi utilizada como controladora da catraca o Arduino Duemilanove com microcontrolador ATmega 328. O servidor de banco de dados foi desenvolvido na aplicação SQL Server da Microsoft. O *software* para controle do servidor de dados e comunicação com as catracas foi todo desenvolvido em linguagem C#, mais precisamente na plataforma .NET também da Microsoft.

A comunicação entre servidor de banco de dados e placa controladora é feita através da *Ethernet*, com protocolo TCP/IP. O projeto ainda possui como componentes da catraca um leitor NFC com comunicação serial RS232, um *display* LCD, um teclado, e um mecanismo de atuação controlados diretamente pelos pinos do processador.

3.1 FUNCIONAMENTO BÁSICO

Inicialmente o usuário chegará na bilheteria e solicitará a entrada no estabelecimento em questão. Então será verificado se o usuário tem ou não cadastro no banco de dados, caso não o mesmo será realizado, caso sim, apenas o acesso será concedido, ambas as ações serão realizadas em uma aplicação específica em C#. Essas informações ficarão armazenadas no banco de dados.

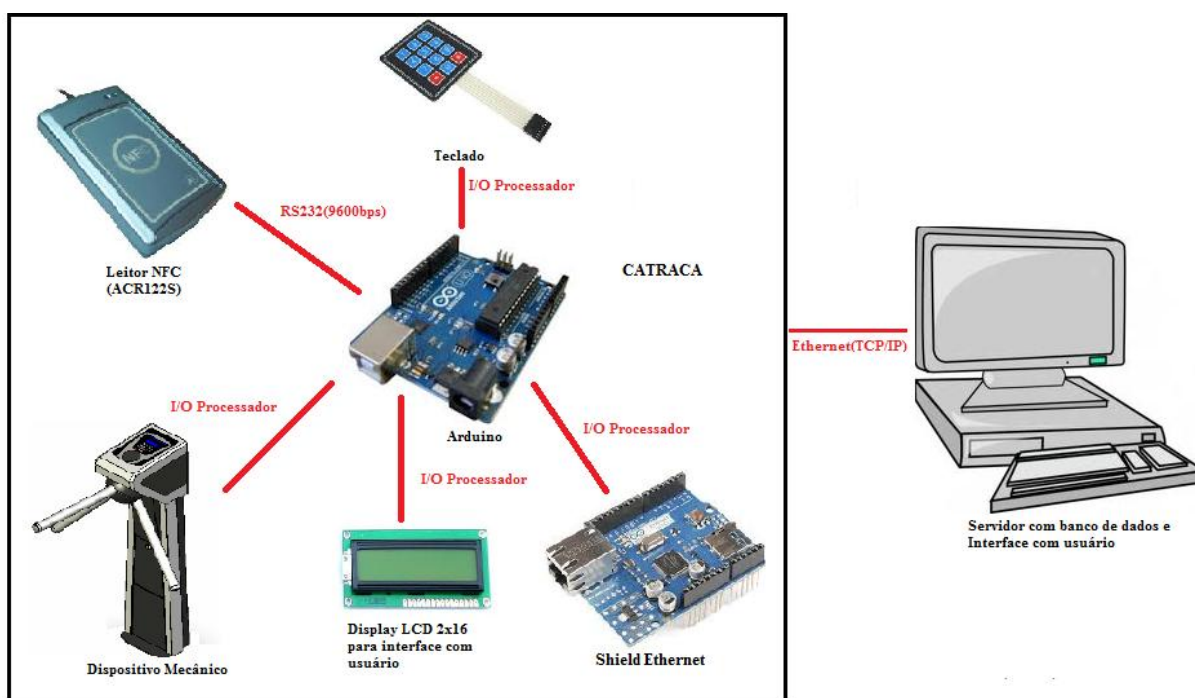


Figura 11– Representação ilustrativa do sistema.

Fonte – Autoria própria

Em seguida, é necessário verificar na própria aplicação C# se as catracas estão conectadas e aptas à comunicação. Caso sim, o mecanismo está apto a receber cartões NFC. Ao receber um cartão NFC, o ID do cartão é transmitido em tempo real para aplicação em C#. Esta acessa o banco de dados, verifica se o cartão existe, e ainda se está com acesso liberado e retorna a informação de acesso liberado, ou acesso bloqueado para controladora. Então a controladora efetua ou não a liberação do mecanismo de acesso. Caso o acesso seja liberado, logo após a controladora receber a informação de acesso liberado, no banco de dados o acesso via o ID em questão é bloqueado de forma a impedir acessos consecutivos.

A figura 12 mostra um fluxograma de funcionamento do sistema desenvolvido.

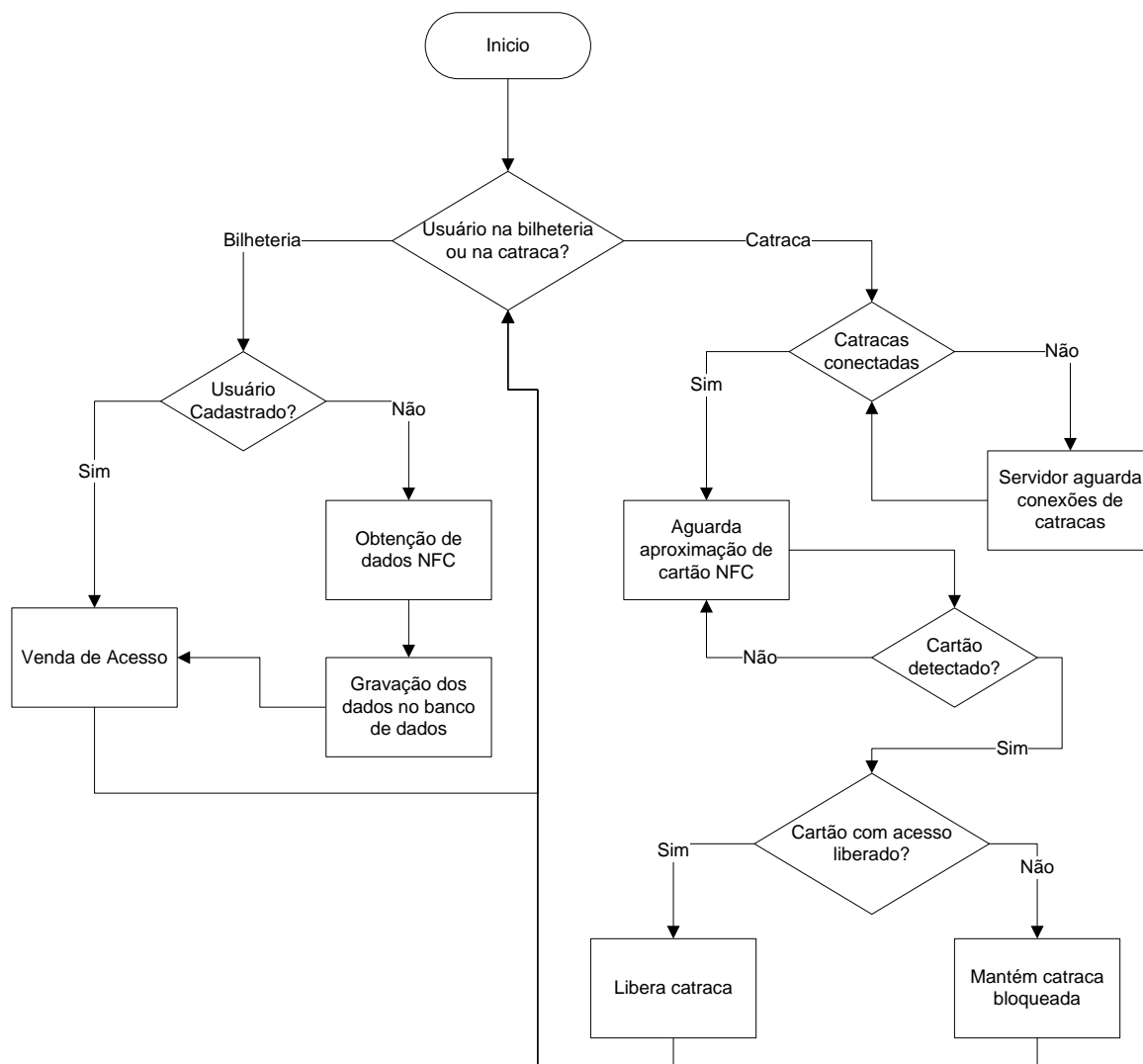


Figura 12– Fluxograma do sistema.

Fonte – Autoria própria

3.2 COMPONENTES DO SISTEMA

3.2.1 Catraca

A catraca é composta pelos seguintes componentes: placa controladora e kit *Ethernet*, leitor de *tags* NFC, mecanismo de atuação, *display* LCD e teclado numérico.

3.2.1.1 Placa Controladora

Para o desenvolvimento desse projeto foi utilizado o kit Arduino Duemilanove, mostrado na figura 13, baseado no microcontrolador ATmega328. “Duemilanove”. Significa 2009 em italiano e o nome foi escolhido pelo ano de lançamento.

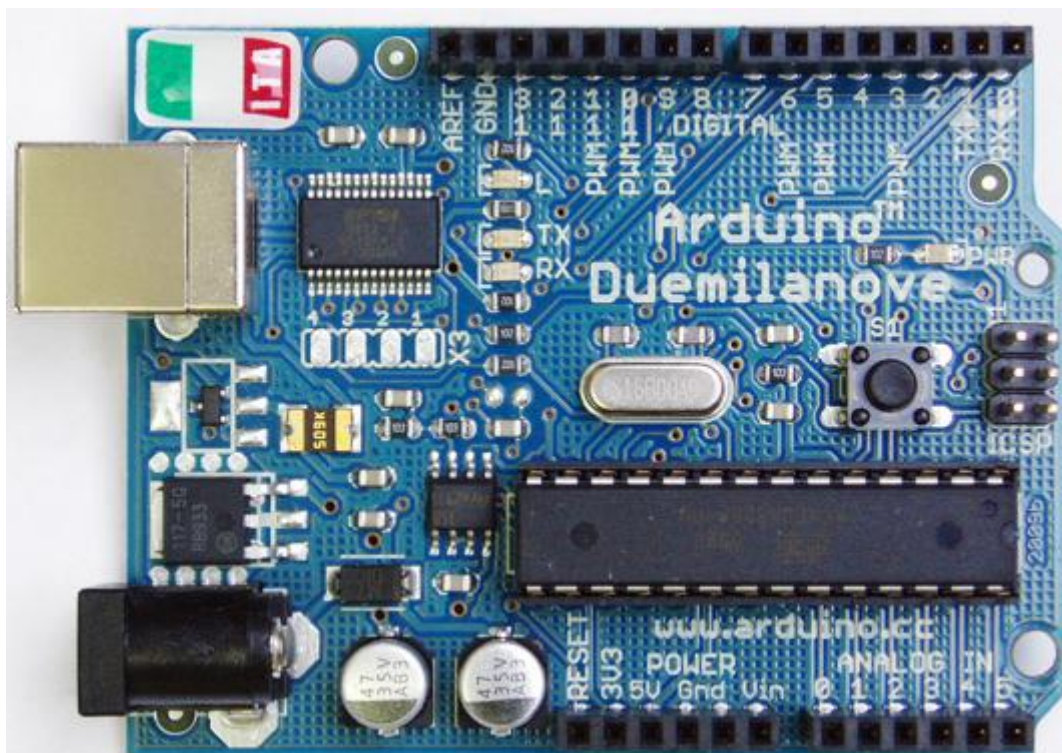


Figura 13– Arduino Duemilanove.

Fonte – Autoria Própria

O Arduino escolhido possui 14 pinos de entrada/saída digital, dos quais 6 podem ser usados como saídas analógicas PWM, 6 entradas analógicas, um cristal oscilador de 16MHz, pinos para receber e transmitir dados seriais TTL, interrupções externas, um LED integrado, uma conexão USB, uma entrada para alimentação e um botão reset. As características do kit escolhido são mostradas na tabela 1.

Tabela 1– Características Arduino Duemilanove

Microcontrolador	ATmega328
Voltagem operacional	5V
Voltagem de alimentação (recomendada)	7-12V
Voltagem de alimentação (limites)	6-20V
Pinos I/O digitais	14 (dos quais 6 podem ser saídas PWM)
Pinos de entrada analógica	6
Corrente contínua por pino I/O	40 mA
Corrente contínua para o pino 3.3V	50 mA
Memória flash	32 kb
SRAM	2 kb
EEPROM	1 kb
Velocidade de clock	16 MHz

Fonte – Aatoria própria

O microcontrolador presente no kit Arduino é o ATmega328 da ATMEL, que é um micro de 8 bits com 32 kb de memória flash para armazenar código, 2 kb de SRAM e 1kb de EEPROM. O esquemático do kit Arduino escolhido é mostrado na figura 14.

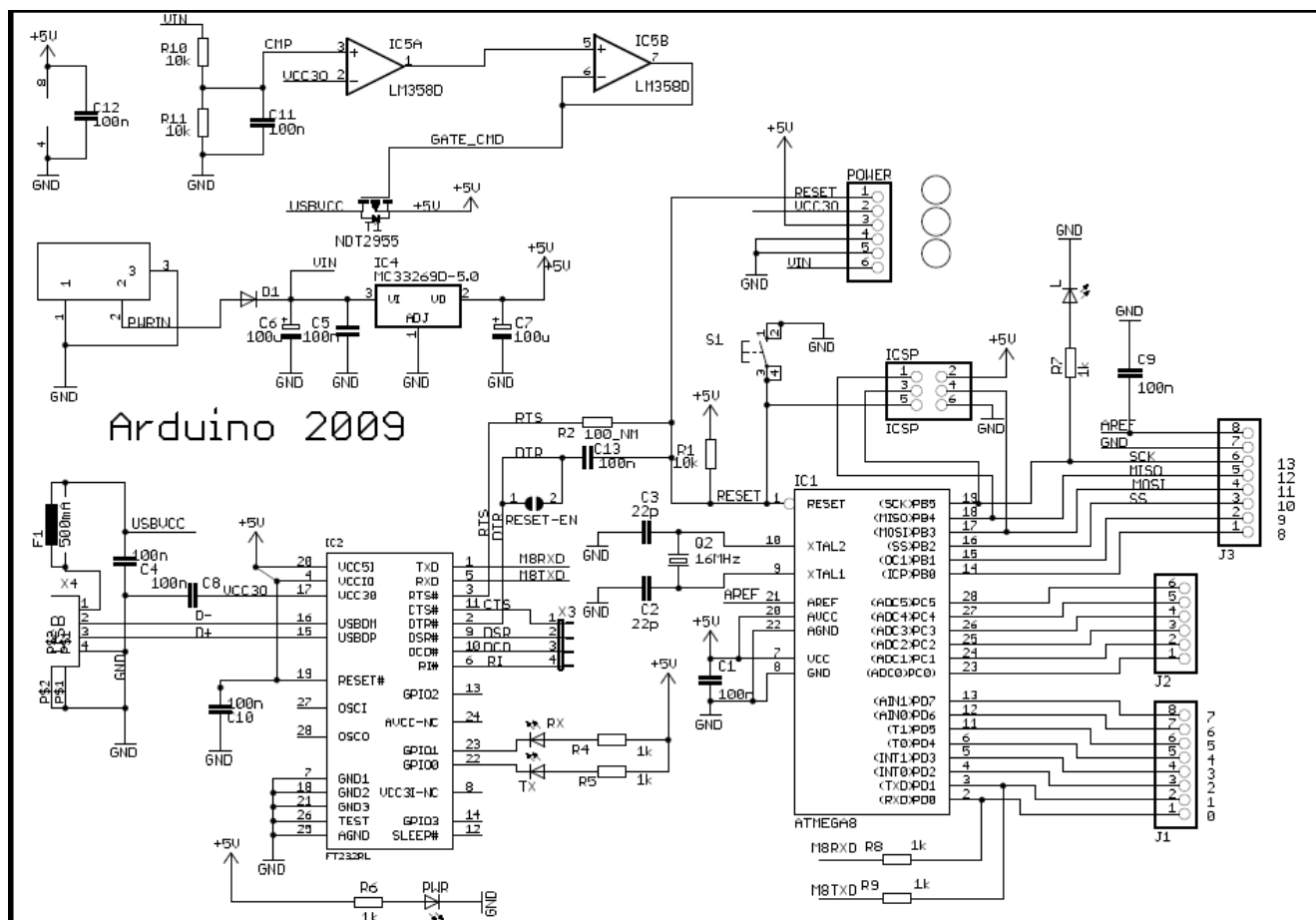


Figura 14 – Esquemático Arduino Duemilanove.

Fonte – (ARDUINO PAGE, 2009)

Foi escolhido esse kit Arduino pelo fato de mesmo facilitar o desenvolvimento, visto que uma série de periféricos já descritos estão presentes. Mesmo assim ainda foi necessário criar um circuito de forma a auxiliar a comunicação serial RS232, comunicação TCP/IP, controle dos solenóides do mecanismo de atuação, conexão com *display* LCD e teclado.

A interface serial RS232 utiliza o circuito integrado MAX232, que recebe e envia ao leitor NFC os dados provenientes do processador. Para o acionamento dos solenóides do mecanismo de atuação foi utilizado um circuito de potência. Para controle do LCD foi utilizado um *shift register*, possibilitando ocupar apenas três pinos de entrada e saída do Arduino. O teclado utilizado é um teclado numérico de quatro linhas e três colunas, necessitando assim de sete pinos de entrada e saída do kit. Por último para comunicação via rede com protocolo TCP/IP é utilizado um *shield ethernet* que possui comunicação serial SPI.

3.2.1.2 Leitor Near Field Communication (NFC)

O leitor NFC utilizado é o ACR122S do fabricante Advanced Card Systems Ltd, mostrado na figura 15.



Figura 15– Leitor ACR122S.

Fonte – Autoria própria

O ACR122S possui interface serial RS232 e alimentação USB. Foi desenvolvido baseado na tecnologia RFID, trabalha na frequência de 13,56 MHz. Suporta Mifare, Felica, e Cartões NFC. É equipado com um buzzer e dois LEDs, para interação com o usuário (ACR122S Communication Protocol).

O leitor escolhido para o projeto tem disponível para acesso documentação como manual, tem também códigos exemplos e demonstrativos escritos em várias linguagens como Java, Delphi 7, Visual C++, Visual C#, e algumas ferramentas que podem ser instaladas em um computador e utilizadas como demonstração das funções do leitor. Uma dessas ferramentas é o ACR122STool, ilustrado na figura 16, em que com ela é possível testar e entender todas as funcionalidades do leitor, como leitura de cartões, utilização dos LEDs e do buzzer.

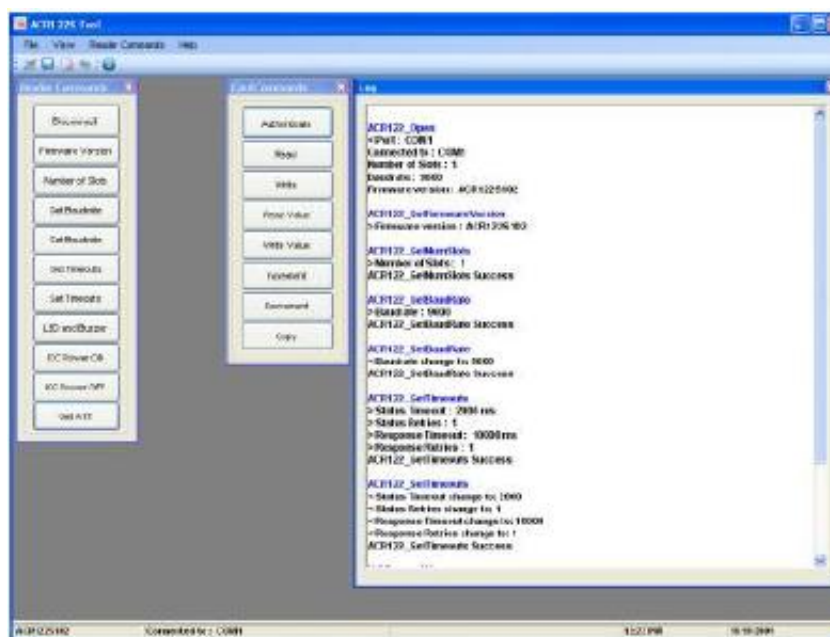


Figura 16– ACR122STool.

Fonte – Autoria própria

O leitor possui uma comunicação serial RS232 com configurações de 9600bps, o protocolo de comunicação tem um padrão um pouco complexo e seus quadros de *bytes* é exibido na figura 17.

Command Frame Format

STX (0x02)	Bulk-OUT Header	APDU Command Or Parameters	Checksum	ETX (0x03)
1 Byte	10 Bytes	M Bytes (If applicable)	1 Byte	1 Byte

Status Frame Format

STX (0x02)	Status	Checksum	ETX (0x03)
1 Byte	1 Byte	1 Byte	1 Byte

Response Frame Format

STX (0x02)	Bulk-IN Header	APDU Response Or abData	Checksum	ETX (0x03)
1 Byte	10 Bytes	N Bytes (If applicable)	1 Byte	1 Byte

Figura 17 – Protocolo de comunicação ACR122S.

Fonte – (ACR122S Communication Protocol)

Basicamente é um *byte* de início, dez *bytes* de cabeçalho, um número variável de *bytes* de comando, um *byte* de soma de verificação (*checksum*) e um *byte* de fim da transmissão.

O *byte* de início por padrão é 0x02. Os *bytes* de cabeçalho indicam o sentido de transmissão e também comandam a interface de cartões, ou seja, ativam ou desativam a mesma no quadro de comando, já no quadro de resposta os *bytes* de cabeçalho apenas indicam qual cabeçalho de comando foi recebido. Os *bytes* de comando ou resposta variam de acordo com o periférico que precisa ser acionado. O *byte* de soma de verificação varia de acordo com cabeçalho utilizado e comando ou resposta recebida. E o *byte* de fim da transmissão por padrão é 0x03. A figura 18 mostra alguns fluxos de protocolo.

1) Activate a SAM

	HOST		RDR
1. HOST sends a frame	→	02 62 00 00 00 00 00 01 01 00 00 [Checksum] 03	
2. RDR sends back a positive status frame immediately		02 00 00 03 (positive status frame)	←
3. RDR sends back the response of the command		.. after some processing delay .. 02 80 0D 00 00 00 00 01 00 00 00 3B 2A 00 80 65 24 B0 00 02 00 82 90 00 [Checksum] 03	←

2) Activate a SAM (Incorrect Checksum, HOST)

	HOST		RDR
1. HOST sends a corrupted frame	→	02 62 00 00 00 00 00 01 01 00 00 [Incorrect Checksum] 03	
2. RDR sends back a negative status frame immediately		02 FF FF 03 (negative status frame)	←
3. HOST sends the frame again.	→	02 62 00 00 00 00 00 01 01 00 00 [Checksum] 03	
4. RDR sends back a positive status frame immediately		02 00 00 03 (positive status frame)	←
5. RDR sends back the response of the command		.. after some processing delay .. 02 80 0D 00 00 00 00 01 00 00 00 3B 2A 00 80 65 24 B0 00 02 00 82 90 00 [Checksum] 03	←

Figura 18 – Fluxo do protocolo de comunicação ACR122S

Fonte – (ACR122S Communication Protocol)

Apesar da documentação fornecida pelo fabricante, entender o protocolo de comunicação não foi uma tarefa fácil. Visto que os *bytes* de cabeçalho e comandos como ler um cartão ou acender o LED, não estavam bem claros na documentação, e ainda nos códigos exemplo, uma parte desses *bytes* estavam mascarados por API do *Windows*. Assim foi necessário partir para uma espécie de engenharia reversa, em que foi conectado o leitor a um computador com o ACR122STool instalado, e ainda entre ACR122STool e o leitor foi colocado um segundo computador com um terminal serial COM aberto, de forma a coletar os dados sendo transmitidos ou recebidos em paralelo, e foi realizado os passos necessários para realizar funções de leitura de cartão, acionar LEDs e buzzer. A figura 19 ilustra a explicação anterior.

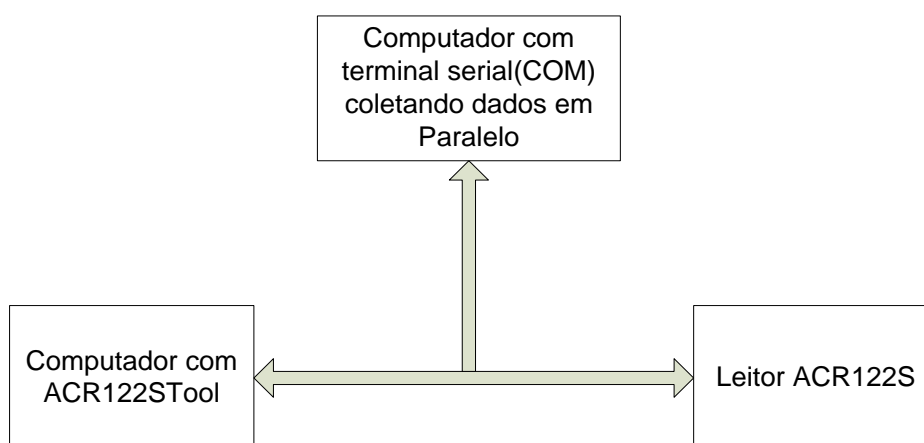


Figura 19 – Leitura de dados da comunicação serial do leitor ACR122S

Fonte – Autoria própria

Adotando o método exposto anteriormente, foram encontrados vários quadros de comando, um deles é mostrado na tabela 2, e o mesmo realiza leitura do ID de um cartão.

Tabela 2 – Quadro com o comando de leitura de um cartão no ACR122S

Início	Cabeçalho	Comando	Soma de verificação	Fim
02h	6F0E0000000004000000h	FF00000009D460010 12023110410h	21h	03h

Fonte – Autoria própria

A partir do quadro mostrado na tabela 2 e a documentação fornecida pelo fabricante, foi possível entender a lógica de cada *byte* utilizado.

O primeiro *byte* é 02h, que indica o início da transmissão. Em seguida temos os *bytes* de cabeçalho. Sendo assim o primeiro *byte* indica a função desejada, nesse caso troca de dados, os próximos quatro *bytes* indicam o número de *bytes* que a parte de comando do quadro terá, o quinto *byte* é zero por padrão, o sexto *byte* é entendido como um número de sequência que será devolvido pelo cabeçalho de resposta, o sétimo *byte* é usado para definir tempos de espera, e por ultimo, os últimos *bytes* são zeros por padrão, isso é ilustrado na figura 20.

Offset	Field	Size	Value	Description
0	bMessageType	1	6Fh	
1	dDwLength <LSB .. MSB>	4	M	Message-specific data length
5	bSlot	1	00-FFh	Identifies the slot number for this command. Default=00h
6	bSeq	1	00-FFh	Sequence number for command
7	bBWI	1	00-FFh	Used to extend the Block Waiting Timeout.
8	wLevelParameter	2	0000h	

Figura 20 – Descrição de cabeçalho de comando de troca de dados.

Fonte – (ACR122S Communication Protocol)

Já o cabeçalho de comando é representado na tabela 3.

Tabela 3 - Descrição do cabeçalho de comando no ACR122S

Bytes	Descrição
FF000000h	Indica transmissão direta de comando
09h	Indica tamanho do comando que virá em seguida
D4600101h	Indica uma transmissão de 212kbps e um alvo passivo
2023110410h	Indica que está sendo pedido o código do sistema (ID)

Fonte – Autoria própria

Da mesma forma que foram encontrados quadros de comando, foram encontrados também quadros de resposta. Um deles é exposto na tabela 4.

Tabela 4 – Quadro com resposta ao comando de leitura de um cartão no ACR122S

Início	Cabeçalho	Resposta	Soma de verificação	Fim
02h	801300000000040100h	D56101100C0100440007048C 3B5A5328809000h	FFh	03h

Fonte – Autoria própria

Sendo assim a figura 21 ilustra a função de cada *byte* do cabeçalho.

Offset	Field	Size	Value	Description
0	bMessageType	1	80h	Indicates that a data block is being sent from the ACR122S
1	dwLength <LSB .. MSB>	4	N	Size of abData field. (N Bytes)
5	bSlot	1	Same as Bulk-OUT	Identifies the slot number for this command
6	bSeq	1	Same as Bulk-OUT	Sequence number for corresponding command
7	bStatus	1		
8	bError	1		
9	bChainParameter	1		

Figura 21 – Descrição de cabeçalho de resposta de troca de dados

Fonte – (ACR122S Communication Protocol)

No quadro de resposta estão vários *bytes*, sendo que o ID do cartão está presente dentro desse quadro. Como pode ser visto na tabela 5.

Tabela 5 - Descrição do cabeçalho de resposta no ACR122S

Bytes	Descrição
D561h	Indica modelo de quadro
01100C0100440007048C3B5A532880h	ID do cartão
9000h	Código de status, nesse caso obteve sucesso

Fonte – Autoria própria

Por último, está presente a soma de verificação e o *byte* de fim da transmissão. O *byte* de fim da transmissão por padrão é 03h, já a soma de verificação é um ou exclusivo (XOR) *byte* a *byte* desde o primeiro *byte* de cabeçalho até o último *byte* de resposta ou comando.

Ainda entre o quadro de comando e resposta, está presente um quadro de quatro *bytes* de status. Esse quadro é capaz de avaliar o quadro de comando e informar se o mesmo está ok, ou tem erro de soma de verificação, erro de tamanho informado, ou se o *byte* que indica o fim da transmissão é diferente de 03h.

Vale ressaltar que com o leitor escolhido é possível realizar muitas outras funções, não aplicadas no projeto, como por exemplo, ler ou escrever dados na memória interna dos *tags* NFC.

3.2.1.3 Shield Ethernet

O Kit Arduino escolhido como placa controladora não possui conexão Ethernet, sendo assim, para realizar o projeto, foi necessário acoplar à placa controladora Arduino Duemilanove um *Shield Ethernet*, de forma que permita o sistema realizar comunicação via socket TCP/IP.

O Kit Ethernet escolhido é o da marca DFRobot mostrado na figura 22, o mesmo é baseado no chip de ethernet Wiznet W5100, que fornece um IP e é capaz de trabalhar tanto em protocolo TCP quanto em UDP.



Figura 22 – Shield Ethernet DFRobot

Fonte – (Arduino Shield List)

Este possui comunicação serial SPI (*Serial Peripheral Interface*), que é um protocolo de dados seriais síncrono. Basicamente a comunicação SPI sempre tem

um *Master*, por exemplo, nesse projeto a placa controladora é o *Master* e o *Shield* o *Slave*. Esta comunicação contém quatro conexões, que se trata de pinos disponíveis no kit Arduino:

- MISO (*Master in Slave OUT*) – Dados do *Slave* para o *Master*.
- MOSI (*Master OUT, Slave IN*) – Dados do *Master* para *Slave*.
- SCK (*Serial Clock*) - *Clock* de sincronização para transmissão de dados entre o *Master* e o *Slave*.
- SS (*Slave Select*) – Seleciona qual *Slave* receberá os dados.

Além dos quatro pinos descritos acima, o *shield* ainda faz uso dos pinos A0 e A1 da placa controladora.

3.2.1.4 Display LCD

Para interface com usuário, escolheu-se para o projeto um display LCD de 16 colunas e duas linhas. Como mostrado na figura 23.



Figura 23 – Display LCD 16x2

Fonte – Autoria própria

Devido às limitações da placa controladora, em relação a número de pinos de entrada e saída digital, não seria viável utilizar um pino para cada bit do barramento de dados, ou seja, ligar diretamente utilizando barramento paralelo. Dessa forma, a solução encontrada foi a utilização do *shiftregister* de 8 bits (precisamente o circuito integrado 74HC595). Esse circuito integrado recebe os dados em barramento serial e os transmite em paralelo.

Utilizando esse *Shiftregister*, pode-se fazer toda a comunicação LCD com processador utilizando apenas três pinos.

3.2.1.5 Teclado

No projeto foi utilizado para interface com o usuário um teclado simples de 16 teclas, porém habilitando apenas 12. A função principal do teclado é entrar com o IP do servidor no sistema de catraca. A figura 24 mostra uma foto do teclado utilizado.



Figura 24 – Foto do teclado utilizado

Fonte – Autoria própria

Foi ligado aos pinos de entrada e saída da placa controladora as quatro linhas e as três colunas da direita, deixando a primeira coluna desabilitada e usando sete pinos de entrada e saída do kit Arduino. Dessa forma o usuário pode entrar com os números correspondentes ao IP do servidor em que o sistema deseja se conectar.

3.2.1.6 Mecanismo de Atuação

O mecanismo de atuação escolhido foi a catraca Wolstar da marca Wolpac, mostrada na figura 25.



Figura 25 – Catraca Wolstar

Fonte – (Wolpac,2011)

Essa catraca possui um mecanismo com solenóide que, ativado eletricamente, libera o giro do braço e, conseqüentemente, o acesso. Há também um sensor de giro, que envia sinais elétricos a um controlador tanto no intuito de

proteger o solenóide quanto detectar quando o usuário passou completamente na catraca.

O solenóide requer corrente de 1,2 ampéres para ativação. Quando presente, essa corrente libera o mecanismo para que haja giro. O sensor de giro conta com dois sensores óticos, os quais a partir da obstrução por um disco metálico com cortes que gira juntamente com os braços da catraca geram pulsos de 3.3V quando a catraca é forçada ou quando seu ciclo de giro chega ao fim.

3.2.1.7 Funcionamento básico da catraca

A primeira tarefa a ser feita para funcionamento da catraca é a escolha entre configuração IP do servidor e o início funcionamento. Caso o IP não esteja corretamente configurado, nesse momento o usuário deve escolher por configurá-lo. Cada controladora deve conhecer o IP do servidor ao qual irá se conectar para transferência de dados. Isto é feito a partir do teclado disponível ao usuário e o IP digitado pode ser visto no *display* LCD. A porta utilizada para conexão é a 80.

Após a configuração do IP a controladora solicita um *reset* no sistema, para validar o novo IP.

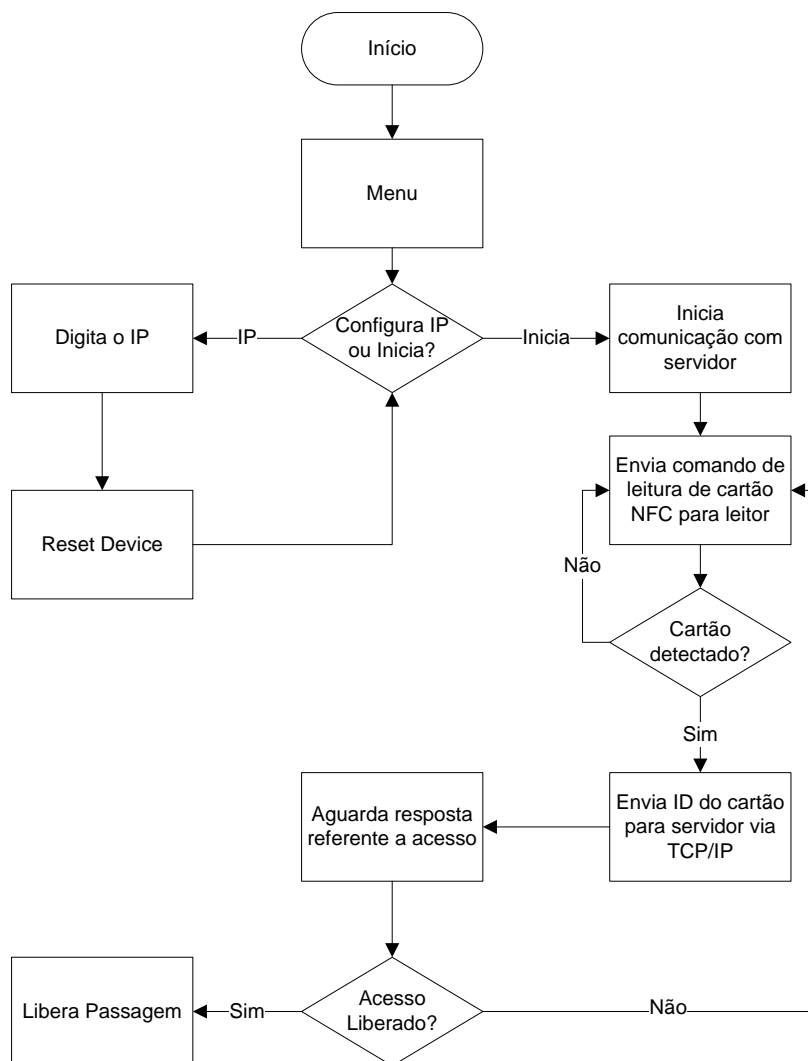


Figura 26 – Funcionamento da catraca.

Fonte – Autoria própria

Logo após o reset o usuário deve escolher pela opção de “Init” no menu. Uma vez iniciada, a catraca passa a aguardar um cartão, caso detecte um cartão a mesma envia ao servidor via TCP/IP o ID do cartão. O Servidor então acessa o banco de dados e verifica se tal ID tem acesso liberado. Caso sim responde a placa controladora uma mensagem, com o caractere “A”, indicando que a passagem pode ser liberada. Caso não responde com uma mensagem, com o caractere “B” indicando que a passagem não pode ser liberada. Logo após receber a mensagem a catraca volta a aguardar a detecção de um cartão.

O funcionamento do sistema referente à catraca é ilustrado na figura 26.

3.2.2 Servidor com banco de dados e interface com o usuário

No que se entende por servidor no projeto está presente o banco de dados e um *software* chamado de Plataforma que faz a ponte entre catraca e banco de dados.

O *software* é uma aplicação que foi desenvolvida em linguagem C# de maneira que seu uso fique o mais intuitivo possível e permita uma interface amigável ao usuário no momento de inserir os dados do cliente no banco de dados, ou ainda verificar ou alterar os mesmos.

A aplicação em C# é capaz de criar o banco de dados em qualquer máquina que a mesma for instalada. O funcionamento se dá através de janelas chamadas *forms*, botões para ações e caixas de texto chamadas *textbox*. É função do *software*, manipular os dados do banco de dados, receber e enviar dados para a catraca via comunicação TCP/IP, e ainda oferecer ao usuário uma interface simples e prática quanto ao funcionamento.

3.2.2.1 Banco de Dados

O banco de dados do projeto foi criado para armazenar as informações dos clientes. Além da identificação do seu cartão NFC, estão contidas informações referentes a nome e RG do cliente. Ainda cada cartão recebe um número, para facilitar a manipulação. O gerenciador foi programado em linguagem SQL. As funções que definem o banco de dados podem ser vistas na figura 27.


	Column Name	Data Type	Allow Nulls
	Cartao	int	<input type="checkbox"/>
	ID	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Nome	nvarchar(MAX)	<input checked="" type="checkbox"/>
	RG	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Acesso	bit	<input checked="" type="checkbox"/>

Figura 27 – Tabela de informações de clientes.

Fonte – Autoria própria

Para cada cliente existe também uma variável chamada “Acesso”, que define se o cliente tem permissão para passagem pelo dispositivo de catraca ou não.

Foram criadas ainda, para facilitar o acesso e manipulação dos dados do banco, algumas *stored procedures* que permitem algumas tarefas como exibir encontrar o nome do cliente através do ID de seu cartão, excluir ou adicionar clientes, entre outras.

No *software* Plataforma, está presente uma classe definida como “BancodeDados” que possui métodos que realiza as funções de manipulação dos dados(figura 28). Assim em qualquer classe do sistema é permitido instanciar um objeto para o banco de dados e manipular os dados do banco a qualquer momento.

Na figura 29 é possível ver um trecho do código para realização do cadastro de clientes.

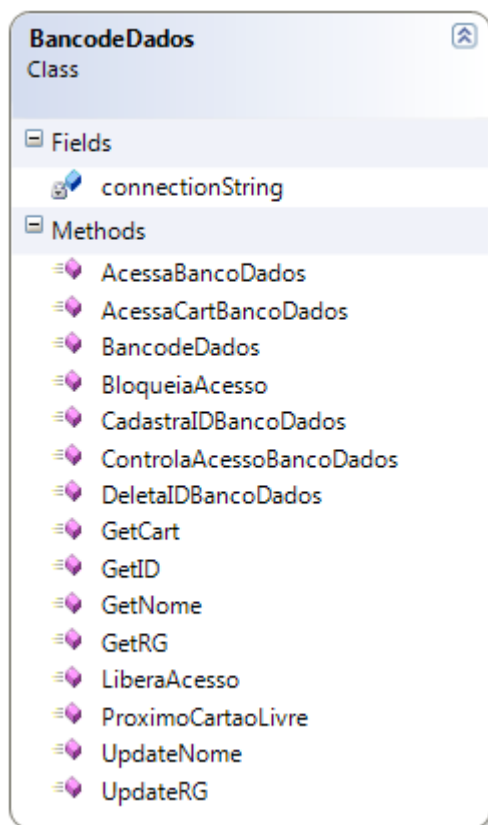


Figura 28 – Tabela de informações de clientes.

Fonte – Autoria própria

```

public void CadastraIDBancoDados(int cart, string ID_msg, string nm,
    bool access, string rg)
{
    Cartao newCart = new Cartao(cart, ID_msg , nm, access, rg);

    string query = "INSERT INTO IDs (Cartao. ID. Nome. RG. Acesso) values
        ( '" + newCart.Cart + "', '" + newCart.ID + "', '" + newCart.Nome
        + "', '" + newCart.Rg + "', '" + newCart.Acesso + "')";

    SqlConnection conn = null;
    conn = new SqlConnection(connectionString);

    if (conn.State == ConnectionState.Closed)
    {
        conn.Open();
    }
    SqlCommand cmd = new SqlCommand(query, conn);

    cmd.ExecuteNonQuery();
    conn.Close();
}

```

Figura 29 – Trecho do código – Cadastrando Clientes

Fonte – Autoria própria

3.2.2.2 Tela principal da Plataforma

Quando o programa é aberto, a classe “Principal” instancia um novo objeto e abre uma janela, apresentada na figura 30, que tem como função, exibir os dados do banco de dados, exibir status de dispositivos conectados, e ainda organizar os seguintes botões: “Cadastrar”, “Excluir”, “Editar”, “Alterar Acesso”, “Ver Dados Recebidos”, “Limpar” e “Reset Server”.

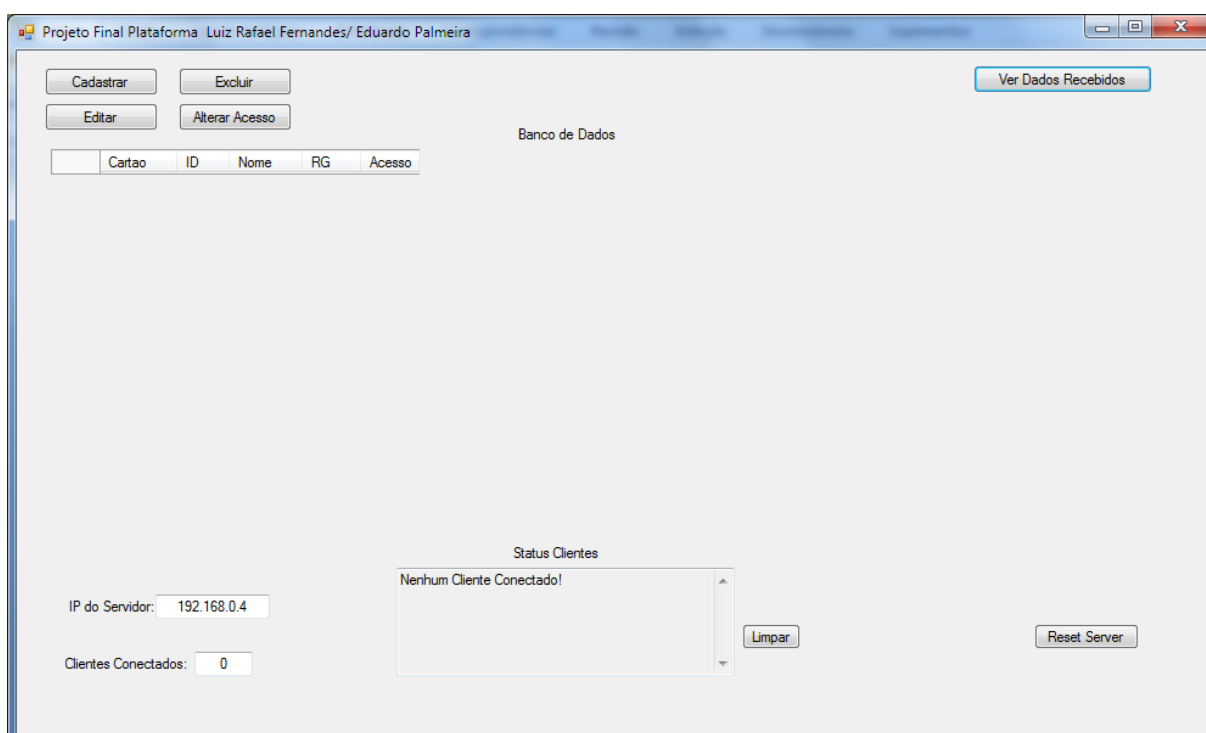


Figura 30 – Tela Principal da Plataforma

Fonte – Autoria própria

O estado das conexões é apresentado na caixa de texto, visivelmente nomeada como “Status Clientes”. Nessa caixa de texto é possível conferir um histórico de conexões de clientes, em que nesse projeto os clientes são sempre catracas, além de o estado atual do cliente, ou seja, informações se o cliente está conectado, ou se já esteve conectado em algum momento, enquanto a aplicação está aberta. Cada cliente é reconhecido pelo seu endereço IP na rede.

Cada vez que um cliente se conecta ao servidor um objeto pertencente à classe cliente é utilizado e o número de clientes conectados informado na caixa de texto do canto esquerdo abaixo da figura 30 é incrementado, para demonstração do projeto foram instanciados apenas três clientes, mas o número de instancias pode ser selecionado de acordo com o número de catracas que estarão presentes no estabelecimento. Para cada cliente instanciado, uma *thread* é criada, e iniciada conforme mostra o trecho de código presente na figura 31, tal thread fica na espera de uma conexão para então receber ou enviar dados. Sendo o programa *multi thread* é possível que vários clientes se conectem e troquem dados simultaneamente com o servidor. Ainda como cada cliente é um objeto da classe Clientes, é possível que o servidor direcione dados apenas para o cliente que ele escolher.

```
public Clientes(Socket m_receiverSock2, Socket m_listenerSock2,
delPutMsg atMsg, delPutMsg atStatus, delPutMsg atGrid, string conexao,
EL_SockServer serv)
{
    connectionString = conexao;
    server = serv;
    m_receiverSock = m_receiverSock2;
    m_listenerSock = m_listenerSock2;
    m_atualizaServerMsg = atMsg;
    m_atualizaServerStatus = atStatus;
    m_atualizaGrid = atGrid;
    m_serverWorkThread = new Thread(EL_TrataConexoes);
    m_serverWorkThread.Start();
    bancodados = new BancodeDados(connectionString);
}
```

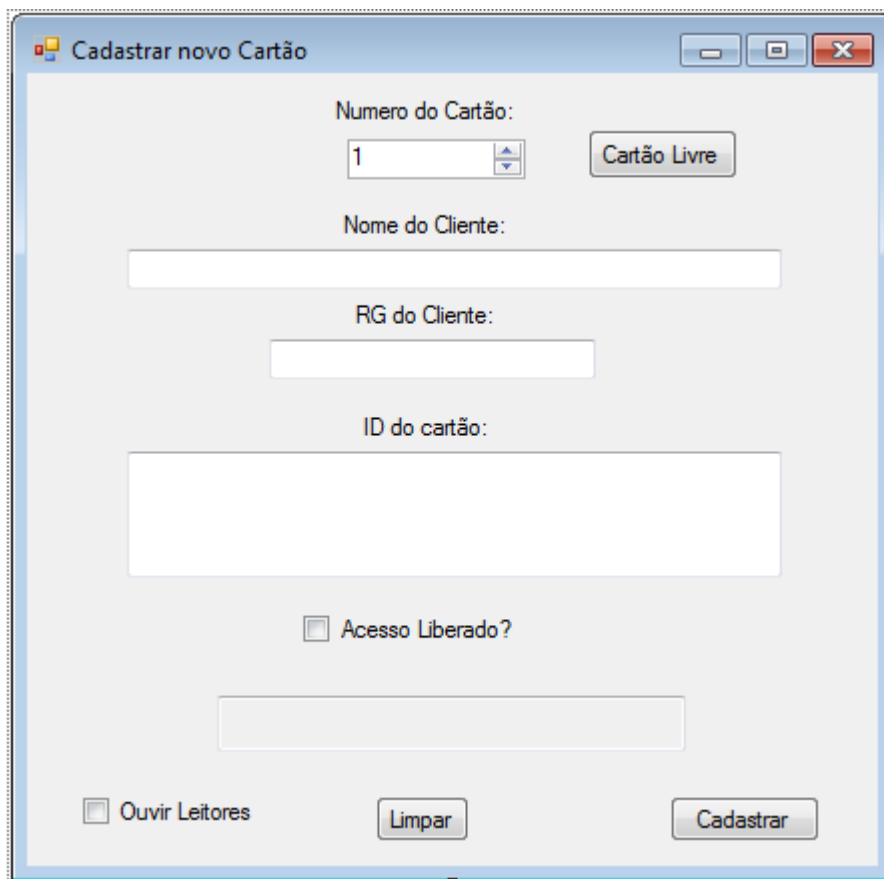
Figura 31 – Trecho do código – Construtora da classe Clientes

Fonte – A autoria própria

Ainda na tela principal da Plataforma está presente o endereço IP do servidor, que servirá de guia para ser inserido nos clientes. Está presente o botão “Limpar” que apenas apaga o histórico de conexões presente na caixa de texto “Status Clientes”. E também estão presentes outros botões que serão discutidos um a um seguir.

3.2.2.3 Cadastrar

Na tela principal está presente o botão cadastrar, que quando acionado exibe uma nova janela, mostrada na figura 32, que é pertencente ao objeto instanciado a partir da classe “FormCadastrar”.



A imagem mostra uma janela de software intitulada "Cadastrar novo Cartão". O formulário contém os seguintes elementos:

- Um campo de entrada rotulado "Número do Cartão:" com o valor "1" e um botão "Cartão Livre" ao lado.
- Um campo de entrada rotulado "Nome do Cliente:".
- Um campo de entrada rotulado "RG do Cliente:".
- Um campo de entrada rotulado "ID do cartão:".
- Um campo de entrada rotulado "Acesso Liberado?" com uma caixa de seleção desativada.
- Um campo de entrada rotulado "Ouvir Leitores" com uma caixa de seleção desativada.
- Dois botões: "Limpar" e "Cadastrar".

Figura 32 – Janela para cadastro de cartões

Fonte – Autoria própria

Nessa janela é possível cadastrar novos cartões. Para isso, o usuário deve preencher os dados: RG, nome do cliente, ID do cartão, *status* de acesso. O usuário deve também informar o estado de acesso para esse cliente, se será imediatamente liberado ou não. É necessário que o número de cartão seja selecionado de tal forma que esse número não esteja sendo utilizado por outro cartão. Caso o usuário sinta dificuldade em encontrar esse número, ele pode clicar no botão “Cartão Livre”, que a Plataforma encontra o número mais baixo disponível.

No projeto existe uma classe identificada por “Cartão”, sendo que para cada novo cartão cadastrado, um novo objeto dessa classe é instanciado. Sendo assim não há limites definidos para o número de cartões que pode ser criado.

Olhando para figura 32 é possível notar um *checkbox* identificado por “ouvir leitores”. Esse, quando marcado possibilita que o próximo cartão NFC lido em uma leitora especificada (que no projeto foi escolhido o da própria catraca por ser o único) tenha seu ID mostrado na caixa de texto correspondente ao “ID do cartão”. Assim é possível cadastrar cada cartão com mais facilidade.

Ainda na janela para cadastro está presente o botão “Limpar” que apenas apaga todos os campos referente a dados do cartão e uma caixa de texto abaixo do *checkbox* referente ao estado de acesso. Essa caixa de texto informa alguns resultados para interação com o usuário, como “cartão cadastrado com sucesso”.

Ao fim da inserção dos dados o usuário deve clicar no botão cadastrar para finalizar o processo. Caso algum dado já esteja presente o mesmo será informado para o usuário e ainda alguma correção será sugerida. Caso tudo esteja certo o cadastro será realizado com sucesso.

3.2.2.4 Excluir

Ao acionar o botão “Excluir” presente na plataforma principal, uma nova janela será aberta, apresentada na figura 33. Essa janela pertence ao objeto instanciado a partir da classe “FormExcluir”.

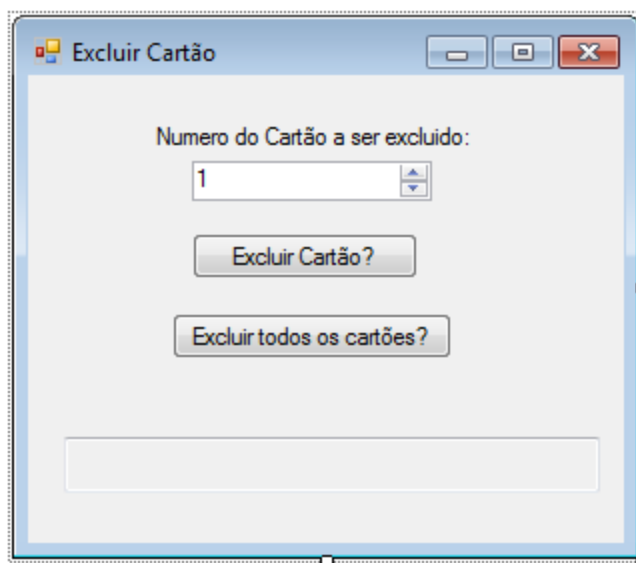


Figura 33 – Janela para excluir cartões

Fonte – Autoria própria

Nessa classe também está presente um objeto da classe “BancodeDados”, sendo assim é possível manipular os dados do banco. Porém nessa janela é possível apenas excluir cartões. Para isso o usuário deve identificar o cartão a ser excluído pelo seu respectivo número e excluir apenas ele, ou ainda poderá excluir todos os cartões. Ainda nessa janela está presente uma caixa de texto, em que o *status* da ação sendo realizada é informado. Na figura 34 é possível ver o trecho do código que a ação de acionar o botão “Excluir Cartão?” provoca.

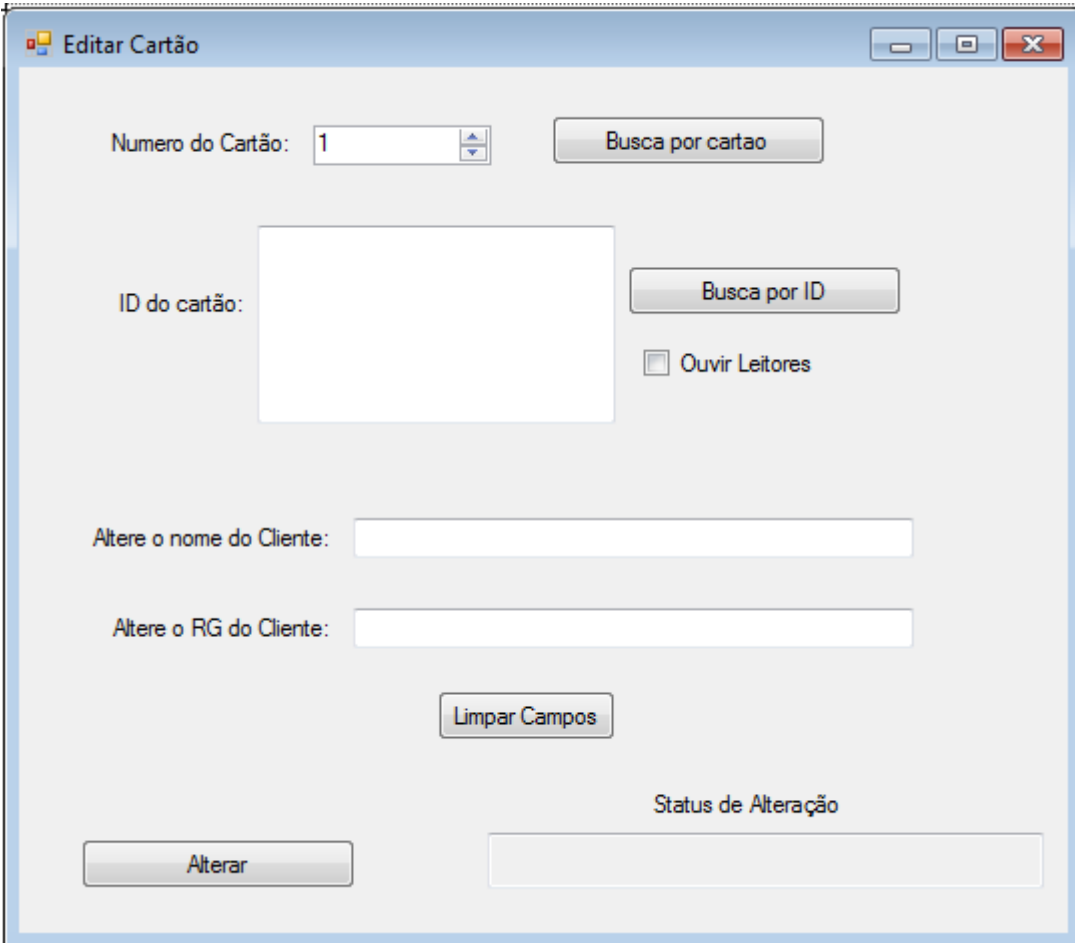
```
private void buttonDel_Click(object sender, EventArgs e)
{
    cart = Convert.ToInt32(DelCaixaTexto.Text);
    if (bancodados.AcessaCartBancoDados(cart) == false)
    { MessageBox.Show("Cartão inexistente, impossivel deletar!", "Erro"); }
    else
    {
        bancodados.DeletaIDBancoDados(cart);
        plataforma.PopulaGrid();
        StatusEditarBox.Text = "Cartao " + cart + " excluído com sucesso!";
    }
}
```

Figura 34 – Trecho código – Botão Excluir Cartão

Fonte – Autoria própria

3.2.2.5 Editar

O botão “Editar” presente na tela principal da plataforma, abre uma nova janela, mostrada na figura 35. Essa janela pertence ao objeto criado a partir da classe “FormEditar”.



A janela "Editar Cartão" possui o seguinte layout:

- Um campo de texto rotulado "Número do Cartão:" contendo o valor "1".
- Um botão "Busca por cartao" à direita do campo de texto.
- Um campo de texto rotulado "ID do cartão:" que está atualmente vazio.
- Um botão "Busca por ID" à direita do campo de texto.
- Um checkbox rotulado "Ouvir Leitores" desativado.
- Dois campos de texto rotulados "Altere o nome do Cliente:" e "Altere o RG do Cliente:", ambos atualmente vazios.
- Um botão "Limpar Campos" centralizado abaixo dos campos de texto.
- Um campo de texto rotulado "Status de Alteração" na parte inferior direita.
- Um botão "Alterar" na parte inferior esquerda.

Figura 35 – Janela para editar cartões

Fonte – Autoria própria

A classe “FormEditar” tem um objeto para classe “BancoDeDados”, sendo assim nessa janela também é possível manipular os dados do banco. Porém, nessa janela é possível apenas editar os dados existentes.

O usuário tem duas formas de encontrar o cartão que deseja alterar no banco de dados. A primeira forma é “Busca por cartão”. Nesse caso o usuário deve digitar o número correto do cartão no campo identificado por “Número do cartão” e

acionar o botão “Busca por Cartão”, então aparecerão os demais atributos do cliente possibilitando o usuário alterá-los. A segunda forma de busca é “busca por ID”, em que ou o usuário digita o ID do cartão da mesma forma que ele digitou o número do cartão já explicado acima, ou o usuário apenas aciona o *checkbox* identificado por “Ouvir Leitores” de forma que o próximo cartão que for aproximado da leitora terá seu ID transferido para o campo ID do cartão, possibilitando ao usuário encontrar o cartão ao qual deseja alterar facilmente, ao fim da correta inserção do ID do cartão o usuário deve clicar no botão “Busca por ID”. Ao fim da alteração deve ser acionado o botão Alterar para realizar o processo.

Também está presente nessa janela uma caixa de texto responsável por informar o status dos processos que estão sendo realizados. O processo é altamente intuitivo, caso algum erro aconteça, o usuário será avisado instantaneamente.

3.2.2.6 Alterar Acesso

Caso o usuário acione o botão “Alterar Acesso” presente na tela principal, a tela apresentada na figura 36 será exibida. Nela é permitido bloquear ou liberar acesso de clientes.

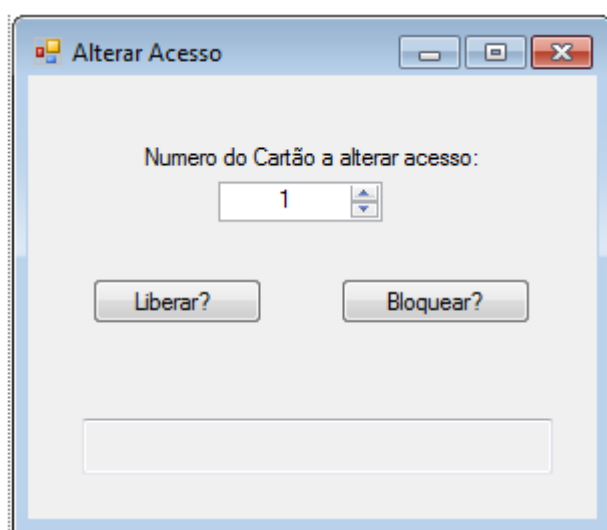


Figura 36 – Janela para alterar acesso de clientes

Fonte – Autoria própria

O usuário deverá selecionar o cartão a partir do numero do cartão, e então clicar no botão “Liberar?” ou “Bloquear” conforme ação desejada. Uma caixa de texto está presente para informar o status do processo ao usuário.

Essa janela pertence à classe “FormAcesso” que possui um objeto para a classe “BancoDeDados” para assim ser possível manipular os dados do banco.

3.2.2.7 Reset Server

Para reiniciar o servidor é necessário que o usuário clique no botão “Reset Server”, esse botão executa o trecho de código apresentado na figura 37.

```
private void button7_Click(object sender, EventArgs e)
{
    m_Ss.StopServer();
    m_Ss = null;

    m_Ss = new EL_SockServer(AtMsg, AtStatusServidor, Grid, connectionString, this);
    StatusClientesBox.Clear();
    StatusClientesBox.Text += "Nenhum Cliente Conectado!" + "\t\t\t\t\n";
    num_cliente = 0;
    NClienteText.Text = num_cliente.ToString();
}
```

Figura 37 – Trecho código – Botão Reset Server

Fonte – Aatoria própria

Em nível de software, ao clicar nesse botão, será excluído o objeto instanciado a partir de classe “SockServer” e instanciado novamente, ou seja reiniciado. Em termos de funcionalidade, todos os clientes conectados serão desconectados, uma vez que o servidor será parado, e logo em seguida poderão se reconectar normalmente. Essa função é importante como uma forma de prevenir qualquer tipo de problema que apareça quanto a conexões durante o funcionamento.

3.2.2.8 Ver dados recebidos

Caso o usuário acione na tela principal o botão “Ver Dados Recebidos” a janela ilustrada na figura 38 será apresentada. A função dessa é monitorar os dados que estão sendo recebidos a partir do cliente.

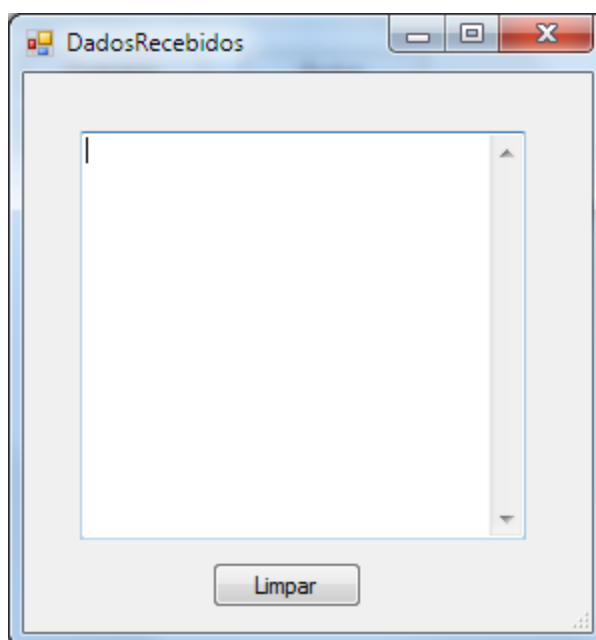


Figura 38 – Janela para monitorar os dados recebidos

Fonte – Autoria própria

Para criar essa janela, existe uma classe nomeada por “DadosRecebidos”, em que na caixa texto presente nessa janela, quando a mesma estiver aberta, é apenas transferido as mensagens recebidos pelos clientes. Caso seja necessário apagar essa caixa texto o usuário poderá clicar no botão “Limpar”.

4 RESULTADOS OBTIDOS

Este capítulo mostra os resultados obtidos ao longo do projeto. Apesar de se tratar apenas de um protótipo, o sistema apresentou funcionamento e confiabilidade necessária para aplicações em situações reais.

4.1 TESTES E VERIFICAÇÕES

A fim de consolidar os conceitos obtidos ao longo do projeto, foi implementado o primeiro protótipo referente aos periféricos integrantes da catraca mostrado na figura 39, com o mesmo foi possível erradicar dúvidas quanto a necessidade de memória *Ram* e *Flash* para o projeto, quanto ao número de pinos de entradas e saídas digitais necessários e, também, quanto ao protocolo de comunicação do leitor NFC.

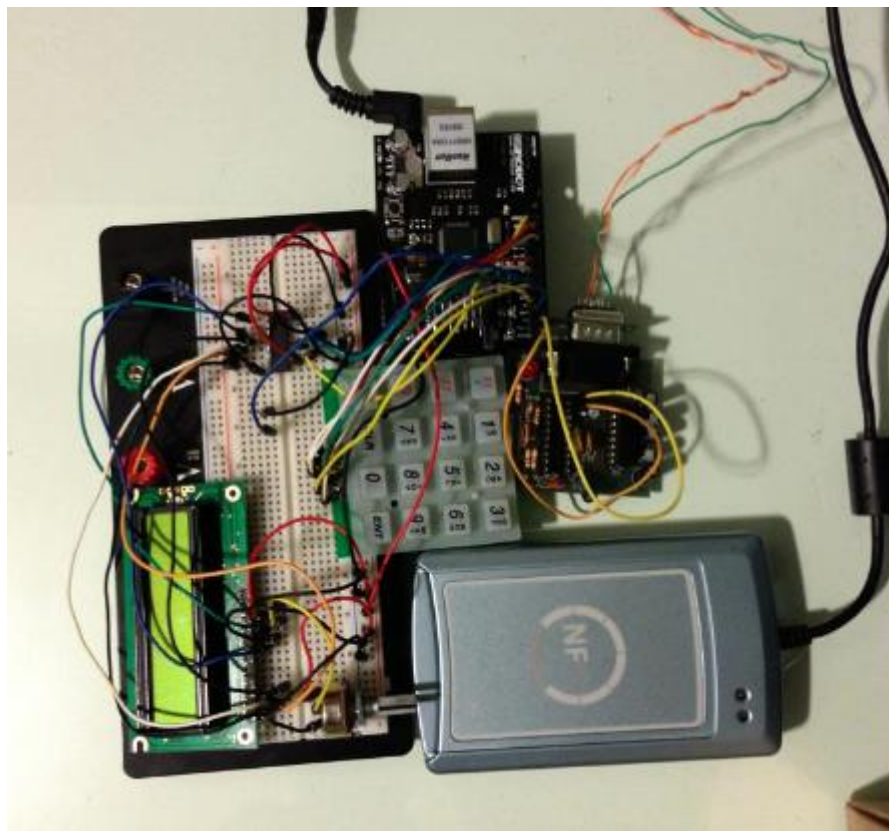


Figura 39 – Primeiro protótipo

Fonte – Autoria própria

Ainda, utilizando o protótipo citado anteriormente, foi possível testar a comunicação via rede, utilizando protocolo *TCP/IP*. Sabe-se que a catraca deve atuar como cliente na rede, então utilizando o *software* Hercules versão 3.2.5, mostrado na figura 40, foi possível simular um servidor para as catracas. Estando esses testes realizados e aprovados, deu-se continuidade ao projeto.

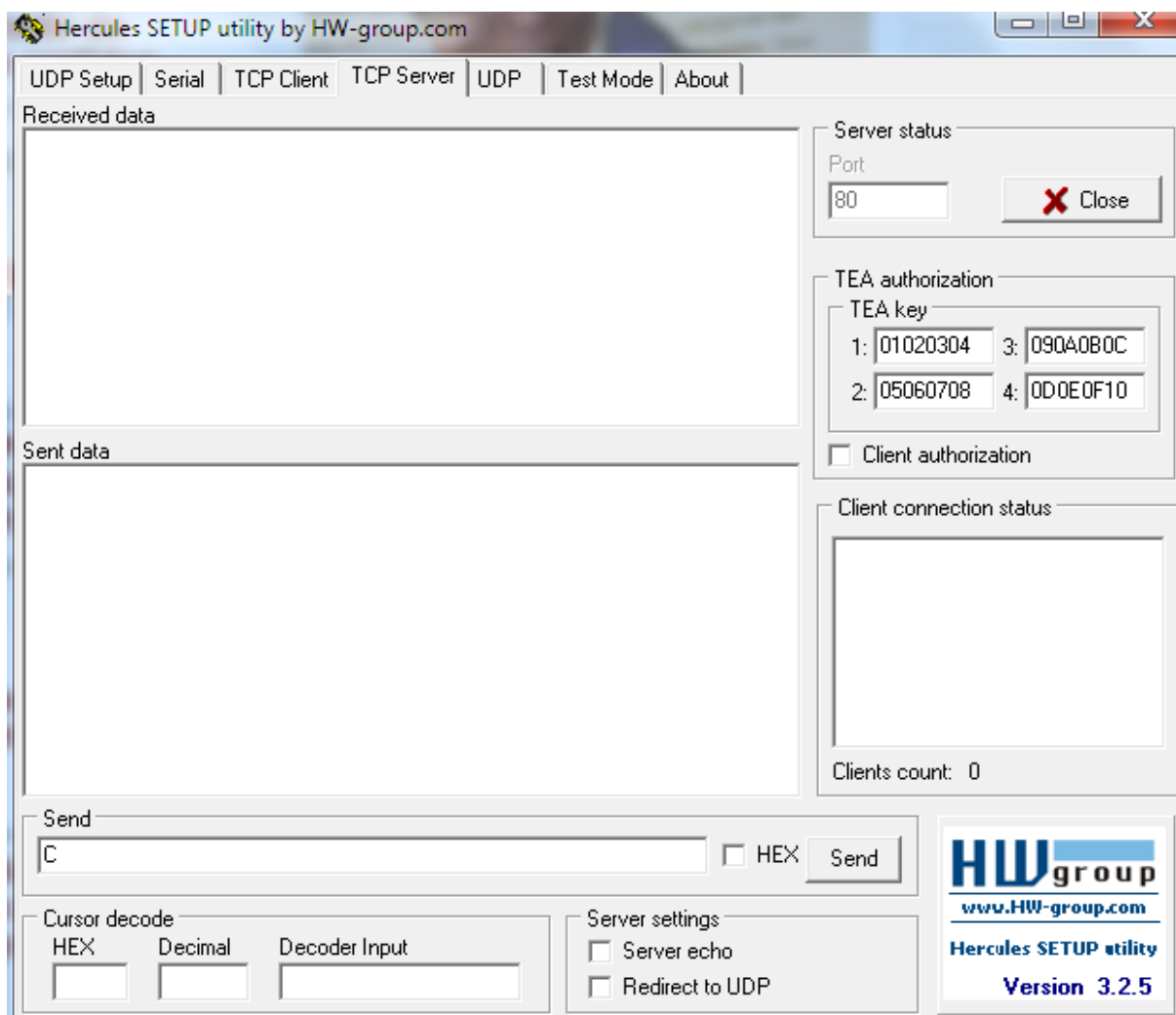


Figura 40 – Software Hercules

Fonte – Autoria própria

Vale ressaltar, ainda que já nessa fase, através de testes de repetibilidade de acesso, utilização de redes diferenciadas e ainda estimativas do tempo de transmissão e recepção de dados, o *hardware* e *firmware* apresentaram confiabilidade necessária para validar a proposta inicial. Partiu-se, então, para confecção do *hardware* final.

Da mesma forma que foram realizados testes e verificações dos componentes que compõem a catraca do projeto, partiu-se, para os testes no *software* gerenciador do banco de dados. Para isso utilizou-se do Hercules mostrado na figura 40, porém o mesmo como cliente, simulando o comportamento esperado da catraca. Assim, foi possível verificar a correta configuração da rede, a manipulação dos dados recebidos e enviados, e ainda o gerenciamento do banco de dados. Ainda, testou-se o *software* desenvolvido em computadores com capacidade computacional inferior, e constatou-se desempenho satisfatório, uma vez que não foram constatadas diferenças significativas no tempo de busca e manipulação dos dados no banco de dados. Com isso, foi finalizado o *software* gerenciador.

4.2 ADMINISTRAÇÃO DO SISTEMA E FUNCIONALIDADES

A aplicação para cadastro e interface com as catracas ficaram em um formato simples, intuitivo e de fácil utilização, minimizando assim um possível tempo de treinamento de usuários e ainda facilitando toda a administração do sistema.

Todos os botões indicam claramente a ação a ser tomada. E ainda em caso do usuário executar o processo de maneira errada uma mensagem informando o motivo do erro e a possível solução muitas vezes é ofertada ao usuário, outras vezes o *software* é capaz de identificar até mesmo a correta ação que o usuário deseja e questionar o usuário se o mesmo não deseja a ação ofertada. O leiaute da aplicação também é claro e simples, diminuindo os erros de operação.

O *menu* da catraca também é altamente intuitivo, ou o usuário entra com o endereço IP do servidor, ou caso o endereço IP esteja correto, apenas inicia o funcionamento normal, autorizando a catraca a tentar conexão com o servidor.

4.3 CONFIABILIDADE

A confiabilidade é a análise de possibilidades de falha do sistema. As falhas mais críticas do sistema se tratam de um cartão com acesso liberado ser rejeitado e vice-versa. A fim de efetuar essa medição foram realizados testes de repetibilidade,

ou seja, foi praticado o acesso utilizando um cartão previamente cadastrado e com livre acesso, aproximadamente mil vezes, chegando a uma taxa de aceitação de aproximadamente 99%. A fim de medir também a taxa de rejeição foi praticado o mesmo procedimento, porém agora com um cartão com acesso bloqueado, chegando a uma taxa de erro de mais ou menos 0,2%. Importante esclarecer que os testes foram realizados tendo no banco de dados cartões com ID diferindo de uma unidade, a fim de ser ter a pior condição. Esses valores são bastante baixos, porém em um banco de dados muito grande, ele pode se tornar significativo.

Ainda, esses erros podem gerar transtornos, como torcedores a mais no estádio, devido a entradas liberadas erroneamente, ou ainda grandes filas de espera devido à falsa rejeição de um cartão.

Há diversas formas de melhorar ainda mais a confiabilidade do sistema. Uma delas seria melhorar o algoritmo de transmissão e leitura dos dados, alterando os atributos do sistema. Também é possível criar setores específicos no estabelecimento a fim de validar entrada por setores.

4.4 CUSTO

O sistema é de baixo custo quando comparado a outros equipamentos similares. A tabela 6 mostra todos os custos para um projeto com uma catraca.

Tabela 6– Custos do sistema

Item	Quantidade	Valor Unit.	Valor Total
Leitor NFC	2	R\$ 150,00	R\$ 300
Mecanismo Catraca	1	R\$ 1.000,00	R\$ 1.000,00
Placa Controladora	1	R\$ 200,00	R\$ 200,00
Componentes	1	R\$ 50,00	R\$ 50,00
Circuito de integração	1	R\$ 30,00	R\$ 30,00
Computador	1	R\$ 1.800,00	R\$ 1.800,00
Adaptador Serial-USB	1	R\$ 30,00	R\$ 30,00
Roteador	1	R\$ 50,00	R\$ 50,00
Licença Software	1	R\$ 1.000,00	R\$ 1.000,00
Valor Total			R\$ 4.460,00

Fonte – Autoria própria

O maior custo é com a compra de computador. Caso o cliente já possuir computadores que sejam capazes de executar a aplicação, o custo se reduz a R\$2660,00.

Mesmo que não exista um sistema idêntico no mercado, muito menos algo com a tecnologia NFC, esse custo é baixo, quando comparado a produtos de funções similares de outros fabricantes. Por exemplo, um sistema de catraca biométrica da marca AKLEMAQ custa R\$3300,00. Porém, a mesma não possui *software* de controle.

5 PLANO DE NEGÓCIOS

5.1 DEFINIÇÃO DO NEGÓCIO

5.1.1 Visão

A empresa RDAccess tem como visão se tornar referência no mercado de sistemas de controle de acesso na America latina, e se posicionar entre as três maiores empresas do setor até 2018.

5.1.2 Missão

Ser um fabricante e prestador de serviços de respeito na America Latina, tornando a marca estável e respeitada. Buscando sempre a inovação, nas mais diferentes maneiras, e assim trazer soluções que resolvam dificuldades no negócio do cliente.

5.1.3 Valores

Os principais valores da empresa RDAccess são:

- Almejar o sucesso com integridade e ética;
- Comprometimento e dedicação ao sucesso dos clientes;
- Qualidade e inovação.

5.1.4 Descrição do Negócio

A empresa visa a venda e prestação de serviços aos clientes. Tais produtos e serviços têm como objetivo principal o controle de acesso de pessoas a eventos

de grande porte. A inovação é outro pilar de alta importância. A busca pelo uso de novas tecnologias, que tornem os produtos mais eficazes e eficientes, é constante. O projeto e montagem, incluindo hardware e software, serão realizados no Brasil.

A empresa visa ter como clientes potenciais grandes e médias empresas, com maior foco em clubes de futebol. Além do produto os clientes poderão firmar contratos de prestação de serviços de manutenção, que garantam a segurança do negócio.

A empresa espera ter como diferencial para o produto a tecnologia NFC (Near Field Communication) que é uma tendência no mercado, e sendo até o momento exclusiva com essa tecnologia. Ainda o serviço oferecido pela empresa para controle do produto, é intuitivo e de fácil aprendizado reduzindo os tempos de treinamento.

5.2 OBJETIVOS

Nesta seção serão apresentados os objetivos da empresa.

5.2.1 Objetivos Principais

A empresa visa desenvolver soluções inovadoras de controle de acesso. Expandir o mercado para a América Latina até 2015. Criar uma filial internacional até 2016. Estabelecer parcerias em outros continentes até 2025.

5.2.2 Objetivos Intermediários

Para alcançar os objetivos principais, são necessários alguns objetivos secundários, tais como:

- Levantar uma lista de 100 clientes possíveis clientes e agendar visitas para demonstração do produto;

- Vender um produto a cada três visitas realizadas;
- Buscar uma equipe de representantes comerciais fora do Brasil;
- Ter uma área de pesquisa e inovação bem estruturada dentro da empresa.

5.3 DEFINIÇÃO DO PRODUTO

5.3.1 Descrição

O produto ofertado é uma solução eletrônica de controle de acesso a estabelecimentos utilizando a tecnologia NFC. Através de aplicações específicas, é possível fazer o cadastro e manipular as catracas de maneira a garantir o acesso dos torcedores às dependências do estabelecimento em questão.

A solução conta com diversas funcionalidades que visam o aumento da segurança e praticidade no acesso a estabelecimentos. O uso do NFC para o acesso promove um controle mais eficaz ao acesso, devido a comunicação ser altamente segura e a possibilidade do cartão NFC ser de posse do usuário, e estar, por exemplo, em seu dispositivo móvel.

Todo o suporte ao sistema será fornecido pelo fabricante. E, em caso de necessidade, haverá a possibilidade de contratação de serviços de manutenção por parte do cliente.

5.3.2 Análise Comparativa

O produto aqui ofertado não possui concorrentes diretos, mas sim produtos e soluções alternativas, pois nenhum concorrente é capaz de oferecer o sistema completo para implantação, por exemplo, em um estádio de futebol. Além disso, não temos sistemas de controle de acesso que utilizem NFC no Brasil. Temos alguns que utilizam, por exemplo, biometria, porém os mesmos possuem limitações quanto ao cadastro, custo, tamanho do banco de dados e facilidade de operação.

Em geral, os sistemas que podem ser comparados são muito limitados quanto ao cadastro, já que todo o banco de dados contendo os dados de cadastro é armazenado na própria catraca. O produto pela empresa ofertado possui uma flexibilidade muito maior que a grande maioria desses sistemas.

O banco de dados não tem muitas limitações. Como o sistema gerenciador de banco de dados é executado em uma máquina de alto desempenho computacional e grande capacidade de armazenamento, não há limitações claras para esse.

Todas as aplicações que gerenciam o sistema foram desenvolvidas pensando na facilidade de uso. São bastante intuitivas e seguras, o que torna o processo de treinamento de usuários para o uso da tecnologia menos oneroso.

5.3.3 Tecnologia

O produto utiliza diversas tecnologias para comunicação, obtenção e armazenamento de dados e controle de acesso físico ao estádio.

O mecanismo de liberação, com solenoide, é baseado no modelo Wolstar fabricado pela Wolpac. Na catraca há uma placa controladora com processador Atmel.

Há uma aplicação para interface com as catracas, cadastro e validação de acesso. A mesma funciona em sistema operacional Windows e foi desenvolvida na plataforma .NET.

O sistema gerenciador de banco de dados é o Microsoft SQL Server. Após a configuração, esse banco de dados é responsável pelo armazenamento de todos os dados dos usuários.

Com a instalação, feita por pessoal devidamente treinado, o usuário facilmente pode operar o sistema, através das aplicações especialmente desenvolvidas para tal.

5.3.4 Produtos e Serviços Futuros

Qualquer sistema de compra fisicamente em estabelecimentos pode gerar filas. Um futuro serviço seria oferecer uma aplicação característica via site de internet em que poderia se adquirir acesso ao estabelecimento e o mesmo ser manipulado no banco de dados de forma a viabilizar a compra do acesso via rede.

5.4 DEFINIÇÃO DO MERCADO

5.4.1 Segmentação de mercado

Todo estabelecimento que comporta eventos de grande porte, precisa de um controle de acesso. E é nesse nicho de mercado que o produto se encontra.

5.4.2 Segmento Alvo

Inicialmente, serão visados clientes de menor porte, ou seja, locais que comportam eventos de menor porte, que desejam modernizar o seu sistema de controle de entrada ao estabelecimento.

Porém, com o tempo, estabelecimentos de maior porte se interessarão na tecnologia e haverá o interesse pela instalação dessas em seus estabelecimentos.

5.4.3 Necessidade

A segurança e o conforto dos usuários que acessam o estabelecimento são essenciais nas dependências do mesmo. E são exatamente essas duas qualidades que o produto ofertado visa. Com a possibilidade de integrar seu dispositivo móvel com o acesso ao estabelecimento a situação fica mais cômoda. Além disso, como

todos os usuários serão cadastrados e aqueles com restrição de acesso não terão acesso, haverá uma maior segurança e controle no estabelecimento.

5.4.4 Tamanho

Todos os estabelecimentos que tenham necessidade de controle de acesso são clientes em potencial. Apesar desse grande mercado, não é necessariamente o grande volumes de vendas que trará grande retorno financeiro, e sim a venda associada a prestação de serviços técnicos associados ao sistema.

5.4.5 Crescimento

Espera-se que a necessidade para serviços de controle de acesso aumente, visto que há uma necessidade constante de aumento de segurança e conforto. Espera-se também que a tecnologia NFC se torne uma tendência nos dispositivos móveis.

5.4.6 Players

Não existe um sistema idêntico no mercado, muito menos algo com a tecnologia NFC. O que tem é produtos similares, como por exemplo, um sistema de catraca biométrica da marca AKLEMAQ custa R\$3300,00. Porém, a mesma não possui *software* de controle. Outra opção é dada pela loja ZAZTecnica. A catraca biométrica, marca TOPLine, tem custo de R\$ 3.850,00. Porém, é necessário ainda um *software* de controle, que pode ser adquirido, no valor de R\$ 1.200,00, ou o cliente pode escolher pagar uma mensalidade de R\$ 60,00.

5.4.7 Modelo de distribuição

A distribuição de equipamentos de controle de acesso segue a lógica de qualquer sistema de controle de acesso eletrônico. É feita intensamente através de empresas que ofereçam o serviço de instalação e representantes. Ainda é possível comprar o produto sem contar com instalação, através de distribuidoras e realizar a instalação a mercê do proprietário do produto.

5.4.8 Modelo de Competitividade

A principal característica que traz competitividade entre as empresas do ramo, é a busca por segurança, controle e praticidade, ao se tratar de acesso a eventos de grande porte.

Aliar essas características a novas tecnologias é o diferencial para alavancar no mercado e tornar um produto competitivo e até líder. Por isso, a pesquisa e desenvolvimento tem um papel de suma importância tratando desse cenário.

O preço final do produto também é relevante. A busca dos clientes pelo melhor preço, aliado com melhor qualidade e atendimento, é umas das áreas que traz competitividade entre as empresas.

5.5 OFERTA E A PROPOSTA DE VALOR

O mercado para a venda do produto se resume a estádios de futebol, e pode ser estendido a qualquer estabelecimento que necessite de controle de acesso. O cliente estabelecido é clubes de futebol de pequeno e médio porte.

A aplicação do produto é um controle de acesso completo a estabelecimentos, oferecendo controle de banco de dados através de aplicação *Windows*, suporte e instalação, utilizando uma tecnologia recente e inovadora no mercado.

O produto tem como atrativo, a tecnologia NFC empregada, trazendo conforto, segurança e praticidade ao usuário. Ainda como diferencial do produto,

aplica-se o fato de não ter solução idêntica no mercado, e ainda nenhuma solução utilizando a tecnologia recente empregada.

5.6 ESTRATÉGIA E IMPLEMENTAÇÃO

As estratégias são pontos fundamentais para que o sucesso seja obtido em uma proposta de valor. Além de estabelecer metas, definir como essas metas serão alcançadas, modifica as vias de se obter a ocorrência da lucratividade.

5.6.1 Diferenciais Competitivos e Proposta de Valor

Para se obter sucesso em um mercado, é necessário um diferencial entre os demais competidores. Esses diferenciais geralmente estão presentes em uma proposta de valor bem elaborada e concisa.

Os principais diferenciais presentes na empresa RDAcces são:

- Valor coerente com o mercado;
- Interface interativa de fácil manuseio;
- Tecnologia acordada com o mercado;
- Praticidade no acesso;
- Tecnologia acompanha a tendência *mobile*;
- Produto agrega segurança e controle do acesso;
- Baixo custo quando comparado a produtos similares;
- Produto inovador;

Esses pontos fortes quando agregados com o diferencial competitivo, deixam a proposta de valor com mais veemência a sua importância no mercado.

5.6.2 Estratégia de Marketing

5.6.2.1 Preço

O cliente comprará o produto e ficará livre da taxa de instalação. Caso seja desejo do cliente, o mesmo pode adquirir serviços personalizados de manutenção e suporte, mesmo dentro do prazo de garantia.

5.6.2.2 Publicidade

A divulgação do produto será feita através dos representantes contratados, que farão o contato inicial com os clientes. Tais representantes serão devidamente treinados para que possam expor aos clientes todas as vantagens do produto.

5.6.2.3 Promoção

Além de facilidades no pagamento, a instalação não terá custo algum para os clientes. O que a empresa procura é o estabelecimento de um vínculo forte com os clientes, onde ambos possam crescer.

5.6.2.4 Instalação

A empresa contará com uma sede bem localizada, em Curitiba. Possivelmente em um bairro próximo ao centro da cidade. Tal sede precisará de três salas de projeto, cada uma com três computadores, uma bancada de testes e uma bancada para montagem de placas. Necessitará de uma sala de reunião, uma sala de recepção e uma sala de gerência.

5.7 ESTRUTURA ORGANIZACIONAL

5.7.1 Estrutura Legal

A empresa se enquadrará no regime de microempresa, e assim tem a possibilidade de aderir ao Sistema Integrado de Pagamento de Impostos e Contribuições das Microempresas e das Empresas de Pequeno Porte, conhecido como Simples/Federal, Lei nº 9.317, de 1996. Na atual legislação, uma microempresa (ME) é a pessoa do empresário, a pessoa jurídica, ou a ela equiparada, que aufera, em cada ano-calendário, receita bruta igual ou inferior a R\$ 240.000,00. A empresa terá capital fechado, logo será dividida e gerenciada apenas pelos sócios iniciais.

5.7.2 Infraestrutura

5.7.2.1 Produção

Como o mecanismo atuador da catraca será comprado separadamente, será necessária apenas a montagem e gravação de firmware do produto. Esses dois procedimentos serão feitos na própria empresa, em sala específica, já que a demanda por catracas não é muito grande. As placas serão montadas por parceiros, empresas de confecção de placas de circuito impresso. Dentro dessa atribuição, existem duas empresas de médio porte em Curitiba, que podem facilmente suprir a necessidade.

5.7.2.2 Fornecedores

As placas de circuito impresso serão projetadas na sede da empresa, e serão produzidas em Curitiba pela PCI-PARANA, CIRCUIBRAS ou CIRCUITEL. O leitor NFC e o processador utilizado serão importados. Os componentes mais

comuns, como cabos e conectores, serão comprados em empresas da cidade, como a BETA COMERCIAL e a PARESLACK.

5.7.2.3 Logística

Todos os produtos que serão adquiridos e fazem parte do sistema, serão entregues na sede da empresa. O sistema será montado dentro da mesma e enviado ao cliente via transportadoras contratadas. Ainda será enviada uma equipe técnica ao local responsável pela instalação do mesmo para o cliente.

5.8 MODELO DE NEGÓCIO CANVAS

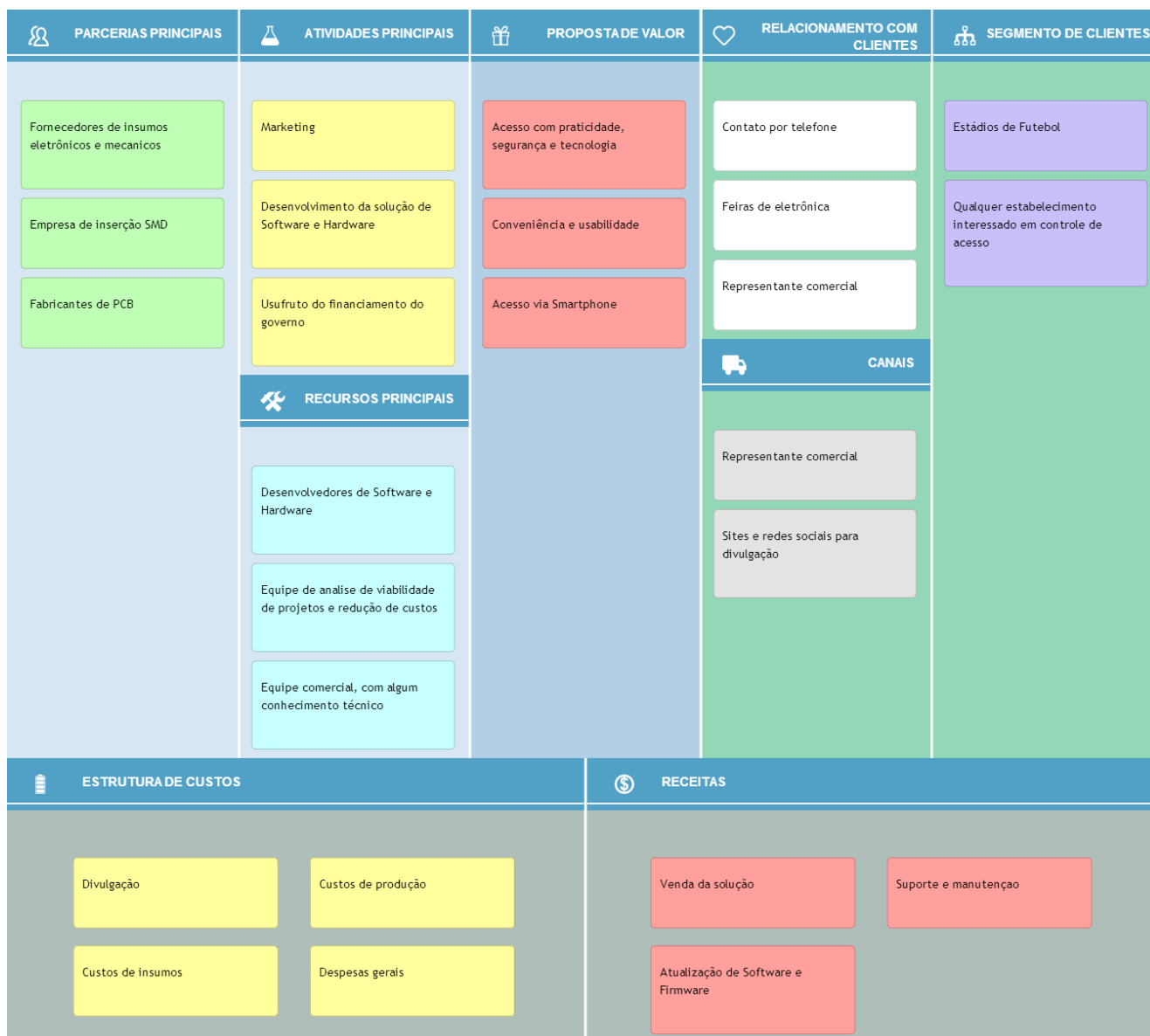


Figura 41 - Modelo de Negócio Canvas da empresa RDAcess

Fonte: Autoria Própria.

A proposta de valor (composta pelas características do produto, interesse do mercado e clientes) é elevada, já que o produto agrega valor com funcionalidades diferenciadas, como: Acesso com praticidade, pois o usuário pode usar seu *smartphone* como passe, controle total dos usuários dentro do estabelecimento, agregando segurança e por ultima utilização intuitiva, diminuindo a necessidade de

treinamento. O cliente disposto a comprar essa solução são clubes de futebol e depois qualquer estabelecimento que necessite de controle de acesso.

Para a venda do produto precisa-se primeiro criar um vínculo de relacionamento com o cliente, para que esse tome conhecimento da RDAccess. Os canais de relacionamento com o cliente são feiras de eletrônica, contato via telefone e internet, e visitas periódicas através de representantes comerciais. A receita da empresa virá das vendas, atualizações do sistema, suporte e manutenção.

Para que o negocio seja consistente, precisam-se realizar algumas atividades consideradas chaves, tais como o desenvolvimento e melhoramento contínuo do produto, investimento em marketing e busca de incentivos e financiamentos do governo. Todos esses gastos estão na estrutura de custos da empresa. E a fim de reduzir os custos, uma atitude que foi tomada foi a busca de parceiros chaves, fidelizando contratos desde que haja uma redução no preço de venda deles.

5.9 PLANO FINANCEIRO

5.9.1 Contabilização

A tabela 7 mostra a contabilização de equipamento e mobiliário da sede da empresa. A tabela 8 mostra a previsão de gastos iniciais

Tabela 7– Equipamentos e Mobiliário da Sede da Empresa

Item	Quantidade	Valor Unit	Valor Total
Mesas de escritório	4	R\$ 2.000,00	R\$ 8.000,00
Cadeiras de escritório	15	R\$ 250,00	R\$ 3.750,00
Luminárias	4	R\$ 100,00	R\$ 400,00
Bancada	3	R\$ 1.000,00	R\$ 3.000,00
Computador	10	R\$ 2.000,00	R\$ 20.000,00
Osciloscópio	3	R\$ 2.200,00	R\$ 6.600,00
Estação de solda	5	R\$ 150,00	R\$ 750,00
Kit de ferramentas	5	R\$ 500,00	R\$ 2.500,00
Impressora	1	R\$ 300,00	R\$ 300,00
Perfuradora	1	R\$ 400,00	R\$ 400,00
Soprador de ar quente	1	R\$ 500,00	R\$ 500,00
Valor Total			R\$ 46.200,00

Fonte – Autoria própria

Tabela 8– Gastos Iniciais

Aluguel de imóvel	R\$ 3.000,00
Instalação de móveis e equipamentos	R\$ 500,00
Pintura	R\$ 850,00
Reparos	R\$ 1.000,00
Limpeza	R\$ 400,00
Seleção e contrato de funcionários	R\$ 1.800,00
Taxas de abertura de Microempresa	R\$ 3.000,00
Material para primeira produção	R\$ 8.000,00
Valor Total	R\$ 18.550,00

Fonte – Autoria própria

5.9.2 Custo

Para o funcionamento da empresa é necessário o levantamento dos custos mensais, mostrado na tabela 9.

Tabela 9 – Gastos Mensais

Item	Quantidade	Valor Unit	Valor Total
Salário Engenheiros	2	R\$ 5.700,00	R\$ 11.400,00
Salário Analistas	2	R\$ 2.000,00	R\$ 4.000,00
Salário Estagiarios	2	R\$ 1.000,00	R\$ 2.000,00
Salário Técnicos	1	R\$ 1.600,00	R\$ 1.600,00
Salário Limpeza	1	R\$ 800,00	R\$ 800,00
Salário Secretária	1	R\$ 800,00	R\$ 800,00
Aluguel	1	R\$ 3.000,00	R\$ 3.000,00
Conta de Luz	1	R\$ 400,00	R\$ 400,00
Conta Telefone/Internet	1	R\$ 150,00	R\$ 150,00
Valor Total			R\$ 24.150,00

Fonte – Aatoria própria

5.9.3 Fluxo de Caixa Esperado

Através de uma análise de expectativa de fluxo de caixa, ilustrado na tabela 10 para os primeiros 12 meses, é possível estimar que com um investimento de R\$200 mil e realizando 10 vendas, a empresa será viável e lucrativa em 18 meses. A estimativa foi feita considerando que em média os clientes pagarão entre 30 e 40 mil reais pelo sistema e mil reais mensais pelos serviços de manutenção. O custo mensal de manutenção da empresa foi calculado e estimou-se em torno de R\$24.150,00.

Tabela 10– Fluxo de Caixa Esperado

Mês		Receitas					Despesas			Acum.
1	1 - Investimento inicial	R\$ 200.000,00				1-Custos iniciais	R\$ 18.550,00			R\$ 181.450,00
2		R\$ 0,00				1- Gastos mensais 2 - Compra de materiais	R\$ 24.150,00	R\$ 30.000,00		R\$ 1.273.000,00
3		R\$ 0,00				1- Gastos mensais 2 - Produção das placas	R\$ 24.150,00	R\$ 1.500,00		R\$ 101.650,00
4		R\$ 0,00				1- Gastos mensais	R\$ 24.150,00			R\$ 77.500,00
5	1 - Primeira Venda	R\$ 30.000,00				1- Gastos mensais	R\$ 24.150,00			R\$ 83.350,00
6	1 - Primeiro contrato de manutenção	R\$ 1.000,00				1- Gastos mensais 2 - Custo de manutenção	R\$ 24.150,00	R\$ 300,00		R\$ 59.900,00
7	1 - Mensalidade do contrato de manutenção 2 - Segunda venda	R\$ 1.000,00	R\$ 40.000,00			1- Gastos mensais 2 - Custo de manutenção	R\$ 24.150,00			R\$ 76.750,00
8	1 - Mensalidade do contrato de manutenção 3 - Terceira venda 2 - Mensalidade do contrato de manutenção	R\$ 1.000,00	R\$ 1.000,00	R\$ 40.000,00		1- Gastos mensais 2 - Instalação do sistema 3 - Custo de manutenção	R\$ 24.150,00	R\$ 800,00	R\$ 300,00	R\$ 93.500,00
9	1 - Mensalidade do contrato de manutenção 3 - Mensalidade do contrato de manutenção 2 - Mensalidade do contrato de manutenção	R\$ 1.000,00	R\$ 1.000,00	R\$ 1.000,00		1- Gastos mensais 2 - Custo de manutenção	R\$ 24.150,00	R\$ 300,00		R\$ 72.050,00
10	1 - Mensalidade do contrato de manutenção 2 - Mensalidade do contrato de manutenção 3 - Mensalidade do contrato de manutenção	R\$ 1.000,00	R\$ 1.000,00	R\$ 1.000,00		1- Gastos mensais 2 - Custo de manutenção	R\$ 24.150,00	R\$ 300,00		R\$ 50.600,00
11	1 - Mensalidade do contrato de manutenção 2 - Mensalidade do contrato de manutenção 3 - Mensalidade do contrato de manutenção	R\$ 1.000,00	R\$ 1.000,00	R\$ 1.000,00		1- Gastos mensais 2 - Custo de manutenção	R\$ 24.150,00	R\$ 300,00		R\$ 29.150,00
12	1 - Mensalidade do contrato de manutenção 2 - Mensalidade do contrato de manutenção 3 - Mensalidade do contrato de manutenção 4 - Quarta venda	R\$ 1.000,00	R\$ 1.000,00	R\$ 1.000,00	R\$ 40.000,00	1- Gastos mensais 2 - Instalação do sistema 3 - Custo de manutenção	R\$ 24.150,00	R\$ 800,00	R\$ 300,00	R\$ 46.900,00

Fonte – Autoria própria

6 CONCLUSÃO

Neste trabalho foi apresentado um sistema de controle de acesso utilizando a tecnologia NFC. Buscando praticidade e segurança, foi mostrado que é possível a utilização da tecnologia NFC, que é visto como uma tendência para os dispositivos móveis, como cartão de acesso para entrada em estabelecimentos.

Não existe no mercado brasileiro ainda soluções que empreguem NFC para esse fim, e ainda o que existe de mais próximo não trata-se de uma solução completa de controle de acesso.

Os principais benefícios deste projeto é a possibilidade de oferecer ao cliente a confiabilidade da tecnologia NFC aliada a praticidade e facilidade, oferecida pela mesma. Além de se tratar de um sistema completo e de fácil utilização.

6.1 CONSIDERAÇÕES SOBRE O DESENVOLVIMENTO DO PROJETO

Partindo do pressuposto que, primeiramente, partiu-se para uma solução inviável, apostando na utilização da placa controladora fornecida pela Wolpac, perdendo assim um bom tempo de desenvolvimento tentando viabilizar outra solução para o projeto. E que o NFC é uma tecnologia totalmente nova. Optou-se pela escolha de um kit controlador simples, para facilitar a comunicação entre processador com leitor NFC.

O *software* e *firmware* foram desenvolvidos em conjunto. Cada código começou com versões bases, passando por fases de testes e aos poucos funcionalidades iam sendo incrementadas. Com isso foi tendo uma evolução gradual até o ponto em que as aplicações contemplam os requerimentos desejados.

O *software* foi totalmente desenvolvido em linguagem C#. Além disso, foram necessários também os estudos sobre banco de dados e linguagem SQL, ambos são conceitos básicos do projeto.

A escolha do kit foi vantajosa, visto que o objetivo era integrar o NFC a uma solução completa de controle de acesso, esse kit possibilitou a simplificação de algumas fases do projeto, tornando a execução do mesmo mais rápido, e possibilitando a adição de vários periféricos.

6.2 OBJETIVOS ALCANÇADOS

O principal objetivo, de utilizar a tecnologia NFC em uma aplicação, foi atingido. Além disso, foi criado um sistema completo de controle de acesso de forma a tornar o mesmo aplicável em estabelecimentos.

A tecnologia empregada foi testada e se mostrou confiável para o propósito. Os periféricos utilizados todos atenderam as especificações. Assim como a placa controladora escolhida para o projeto. Porém, a placa escolhida para o projeto, faz parte de um kit para desenvolvimento, seria necessário, pensando em versão comercial, migrar todo o projeto para um novo processador, especificado para atender os requisitos do projeto.

A aplicação para cadastro, interface com o usuário e interface com as catracas, é intuitiva e confiável. Apresenta um bom desempenho, mesmo em máquinas de menor capacidade computacional.

O tempo total de busca e validação no banco de dados é muito pequeno, e não varia de maneira significativa de forma a prejudicar o desempenho do sistema.

Todas as funcionalidades implementadas no projeto, apresentaram um bom desempenho. Sendo assim apesar de se tratar de um protótipo, o mesmo se trata de uma solução viável para o mercado.

REFERÊNCIAS

ACR122S. ACR122S Communication Protocol. Versão 2.0. Disponível em: <http://www.acs.com.hk/drivers/eng/CMP_ACR122S_v2.00.pdf>. Acesso em 14/07/2013

ALFORD, Ryan. C# Corner. Communicating with Serial Port in C#. Dezembro de 2008. Disponível em: <<http://www.csharpcorner.com/uploadfile/eclipsed4utoo/communicating-with-serial-port-in-C-Sharp>>. Acesso em 29/06/2013

ARDUINO. Arduino home Page. Disponível em: <<http://arduino.cc/en/uploads/Main/arduino-duemilanove-schematic.pdf>>. Acesso em 13/07/2013.

ATOJI, Rodolpho Iemini. Bluetooth e NFC: estudo de caso. Dezembro de 2010. Disponível em: <<http://www.ime.usp.br/~cef/mac499-10/monografias/rodolpho/pdf/mac499-monografia.pdf>> Acesso em 22/06/2013.

ENGENHARIA. Departamento. Manual Técnico Catraca Wolstar II. Março de 2011. Revisão 01.

KIOSKEA .net. Os Modelos de SGBD. Disponível em: <<http://pt.kioskea.net/contents/bdd/bddtypes.php3> >. Acesso em 24/06/2013.

LEVANDOVSKI, Fausto. O método de comunicação NFC e sua aplicação no processo de pagamento através de dispositivos móveis. Junho de 2011. Disponível em: <<http://www.faustolevandoski.com.br/downloads/Near%20Field%20Communication.pdf>> Acesso em 22/06/2013.

MACHINE.Receipt. Adventure with my Arduino. Arduino LCD with only 3 pins. Janeiro de 2013. Disponível em: <<http://receiptmachine.wordpress.com/2013/01/22/arduino-lcd-display/>>. Acesso em 15/07/2013.

MARQUES, Lílian. G1 BA. Março de 2013. Disponível em: <<http://g1.globo.com/bahia/noticia/2013/03/pm-usa-bomba-de-efeito-moral-durante-venda-de-ingressos-do-ba-vi.html>>. Acesso em 11/06/2013.

MSDN. Microsoft. Getting Started with the .NET Framework. Disponível em: <<http://msdn.microsoft.com/en-us/library/hh425099.aspx>>. Acesso em 29/06/2013.

MSDN. Microsoft. Introdução à linguagem C# e ao Framework .NET. Disponível em: <[http://msdn.microsoft.com/pt-br/library/z1zx9t92\(v=vs.90\).aspx](http://msdn.microsoft.com/pt-br/library/z1zx9t92(v=vs.90).aspx)>. Acesso em 29/06/2013

MSDN. Microsoft. Introdução à plataforma .NET da Microsoft. Disponível em: <<http://msdn.microsoft.com/pt-br/aa702903.aspx>>. Acesso em 29/06/2013

NAVATHE, Shamkant B.; ELMASRI, Ramez. FUNDAMENTALS OF DATABASE SYSTEMS 4.ed. Pearson, 2004.

NXP. Semiconductors. Datasheet 74HC595. Revisão 6. Dezembro de 2011. Disponível em: <http://www.nxp.com/documents/data_sheet/74HC_HCT595.pdf>. Acesso em 15/07/2013.

PERON, Marluce. Tecmundo. Julho de 2009. Disponível em: <<http://www.tecmundo.com.br/camera-digital/2431-novas-tecnologias-near-field-communication.htm>>. Acesso em 11/06/2013.

SHIELD LIST. Arduino Shield List. DFRobot Ethernet Shield. Disponível em: <<http://shieldlist.org/dfrobot/ethernet>> Acesso em 15/07/2013.

SILVA, César. Fevereiro de 2013. Disponível em: <<http://clubetelemoveis.com/samsung-e-visa-pretendem-acelerar-os-pagamentos-movéis-via-nfc/>> Acesso em 23/06/2013.

TAKAI, Osvaldo Kotaro. Introdução a banco de dados. Fevereiro 2005. Disponível em: <<http://www.ime.usp.br/~jef/apostila.pdf>>. Acesso em 25/06/2013.

WOLPAC. Catracas pedestais. Disponível em: <http://www.wolpac.com.br/admin/galeria/info/wolpac_1325870398.pdf>. Acesso em 16/07/2013.

ZANNI, Marco. Gizmodo. Fevereiro de 2013. Disponível em: <<http://gizmodo.uol.com.br/pagar-com-celular-e-isso-tudo-ou-estao-tentando-empurrar-o-nfc-goela-abaixo/>> Acesso em 12/06/2013.