

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA  
ENGENHARIA INDUSTRIAL ELÉTRICA – ÊNFASE EM ELETRÔNICA E  
TELECOMUNICAÇÕES**

**ABNER KLOSS KAMAROSKI  
LUIZ ANTONIO BARDELLI**

**PROTÓTIPO DE SISTEMA DE CATRACAS BIOMÉTRICAS PARA  
ESTÁDIO DE FUTEBOL**

**TRABALHO DE CONCLUSÃO DE CURSO**

**CURITIBA  
2013**

**ABNER KLOSS KAMAROSKI  
LUIZ ANTONIO BARDELLI**

**PROTÓTIPO DE SISTEMA DE CATRACAS BIOMÉTRICAS PARA  
ESTÁDIO DE FUTEBOL**

Trabalho 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 de Eletrônica, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Juliano Mourão Vieira

**CURITIBA**

**2013**

Folha destinada à inclusão do **Termo de Aprovação** (elemento obrigatório) a ser fornecido pela Secretaria do Curso ao qual se está vinculado. Recomenda-se digitalizar e inserir nesta folha o termo. Atenção: deverá conter a assinatura de todos os membros da banca examinadora.

Dedicamos este trabalho a nossas famílias por toda a compreensão e apoio que tiveram nesses nossos anos dedicados à universidade.

## RESUMO

KAMAROSKI, Abner Kloss; BARDELLI, Luiz Antonio. **Protótipo de sistema de catracas biométricas para estádios de futebol**. 2013. 69 páginas. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Industrial Elétrica com Ênfase em Eletrônica e Telecomunicações) - Universidade Tecnológica Federal do Paraná. Curitiba.

O objetivo deste projeto é o desenvolvimento do PSCBEF – Protótipo de Sistema de Catracas Biométricas para Estádios de Futebol – um sistema de cadastro e controle de acesso de torcedores que utiliza biometria, mais especificamente impressões digitais. A proposta deste trabalho é apresentar um projeto de controle de acesso dos torcedores às dependências do estádio, buscando maior segurança para todas as pessoas presentes. Para tal, antes do acesso às arquibancadas do estádio será necessário um cadastro dos torcedores que contém a impressão digital, dados pessoais e uma foto. O projeto é dividido em quatro pilares: um sistema de banco de dados, uma aplicação de cadastro, as catracas e um servidor para essas. Tanto a comunicação das catracas com o servidor de catracas quanto a dos computadores de cadastro com o servidor são feitas através de rede *ethernet* com cabos padrão UTP, através do protocolo TCP/IP. As aplicações baseiam-se na linguagem de programação C#, o firmware na linguagem C, e todo o banco de dados na linguagem SQL.

**Palavras-chave:** Biométrico. Digital. Torcedor. Cadastro

## ABSTRACT

KAMAROSKI, Abner Kloss; BARDELLI, Luiz Antonio. **Protótipo de sistema de catracas biométricas para estádios de futebol.** 2013. 69 páginas. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Industrial Elétrica com Ênfase em Eletrônica e Telecomunicações) - Universidade Tecnológica Federal do Paraná. Curitiba.

The purpose of this work is the development of PSCBEF - Prototype System for Football Stadiums Turnstiles Biometrics - a system for fan registration and access control which uses biometrics, specifically fingerprints. This project aims to control the fans that should have access to the premises of the stadium, resulting in greater security for all viewers. To this end, before having access to the bleachers of the stadium, the supporters will need registering which contains the fingerprint, a photo and personal data. The project can be divided primarily on the development of four pillars: a system database, an application for registration, the turnstiles themselves and a server for these. The communication with the server and the computer registration with the server are done via standard Ethernet cables with UTP, using TCP / IP. The applications are based on the programming language C#, the firmware on C, and all databases in SQL.

**Keywords:** Biometric. Fingerprint. Fans. Register.

## Lista de Figuras

Figura 1 - Impressões digitais do tipo “arch” em (a), “loop” em (b) e “whorl” em (c)..	16
Figura 2 - Minúcias do tipo: “ridge ending” em (a), “bifurcation” em (b) e “short ridge” em (c).....	18
Figura 3 - Exemplo de associação de vetores as impressões digitais. ....	19
Figura 4 - Possibilidades de performance. ....	19
Figura 5 – Representação do Sistema .....	24
Figura 6 - Fluxograma de funcionamento.....	25
Figura 7 - LPCXpresso BaseBoard .....	27
Figura 8 - Diagrama esquemático da placa de interface .....	27
Figura 9 - Sensor biométrico SFM3520 do fabricante Suprema.....	28
Figura 10 - Estrutura do pacote de comunicação.....	29
Figura 11 - Pacote da Imagem .....	30
Figura 12 - Catraca Minibloqueio da marca Wolpac.....	30
Figura 13 - Funcionamento da catraca.....	32
Figura 14 - Tela Principal da Aplicação de Cadastro .....	34
Figura 15 - Tela de Cadastro do novo Torcedor.....	34
Figura 16 - <i>Textbox</i> .....	35
Figura 17 - <i>RadioButton</i> .....	36
Figura 18 - Janela de captura de imagem .....	36
Figura 19 - Janela da captura da impressão digital.....	37
Figura 20 - Trecho de código - Botão Salvar e Sair .....	38
Figura 21 - Edição do cadastro .....	39
Figura 22 - Excluir Torcedor .....	40
Figura 23 - Validação da entrada do torcedor .....	41
Figura 24 - Montagem da impressão.....	42
Figura 25 - Impressão resultante.....	43
Figura 26 - Janela de Autenticação.....	44
Figura 27 - Funcionamento básico do aplicativo de interface .....	45
Figura 28 - Diagrama de classes da aplicação de interface com as catracas .....	47
Figura 29 - Aplicação de interface com o usuário .....	49
Figura 30 - Tabela de informações de torcedores.....	51

## Lista de Tabelas

Tabela 1 - Velocidade de comparação de impressões digitais.....	47
Tabela 2 - Confiabilidade do sistema. ....	48
Tabela 3 - Custos da Catraca.....	54
Tabela 4 - Mobiliário e Equipamento da Sede da Empresa. ....	61
Tabela 5 - Gastos Iniciais. ....	62
Tabela 6 - Gastos Mensais.....	62
Tabela 7 - Fluxo de caixa. ....	64



## Lista de Siglas

- PSCBEF – Protótipo de Sistema de Catracas Biométricas para Estádios de Futebol
- CBF – Confederação Brasileira de Futebol
- RFID - *Radio-Frequency Identification* (Identificação por radiofrequência)
- TRR - *True Rejection Rate* (Taxa de Rejeição Verdadeira)
- FAR - *False Acceptance Rate* (Taxa de falsa aceitação)
- FRR - *False Rejection Rate* (Taxa de falsa rejeição)
- TAR - *True Acceptance Rate* (Taxa de falsa aceitação)
- BD - Banco de dados
- SGBD - Sistemas Gerenciadores de Banco de Dados
- SQL - *Structured Query Language* (Linguagem de Consulta Estruturada)
- CLR - *Common Language Runtime* (Ambiente de Execução Independente de Linguagem)
- IDE - *Integrated Development Environment* (Ambiente Integrado de Desenvolvimento)
- JTAG - *Joint Test Action Group* (Grupo de Ação de Teste Conjunta)
- USB - *Universal Serial Bus* (Barramento Serial Universal)
- LED - *Light Emitting Diode* (Diodo Emissor de Luz)
- SPI - *Serial Peripheral Interface* (*Interface Periférica Serial*)
- UART - *Universal Asynchronous Receiver/Transmitter* (Receptor/Transmissor Assíncrono Universal)
- IP – *Internet Protocol* (Protocolo Internet)
- TCP - *Transmission Control Protocol* (Protocolo de controle de transmissão)

## Sumário

<b>1 INTRODUÇÃO .....</b>	<b>12</b>
<b>1.1 CONTEXTUALIZAÇÃO .....</b>	<b>12</b>
<b>1.2 MOTIVAÇÃO E JUSTIFICATIVA DO PROJETO .....</b>	<b>12</b>
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>14</b>
<b>2.1 BIOMETRIA .....</b>	<b>14</b>
2.1.1 Impressões Digitais .....	14
2.1.1.1 Sensores de Impressão Digital .....	15
2.1.1.2 Identificação Através da Impressão Digital .....	16
2.1.1.2.1 Método das Minúcias .....	17
2.1.1.2.2 Método da Correlação.....	18
2.1.1.2.3 Confiabilidade.....	19
<b>2.2 BANCO DE DADOS .....</b>	<b>20</b>
2.2.1 Sistemas Gerenciadores de Bancos de Dados .....	20
2.2.1.1 Modelos de Sistemas Gerenciadores de Bancos de Dados.....	21
2.2.2 Linguagem SQL .....	21
<b>2.3 Plataforma .NET.....</b>	<b>22</b>
2.3.1 Linguagem C# .....	23
<b>3 DESENVOLVIMENTO DO PROJETO.....</b>	<b>23</b>
<b>3.1 FUNCIONAMENTO BÁSICO.....</b>	<b>24</b>
<b>3.2 COMPONENTES DO SISTEMA .....</b>	<b>26</b>
3.2.1 Catracas .....	26
3.2.1.1 Placa Controladora.....	26
3.2.1.2 Sensor Biométrico .....	28
3.2.1.3 Mecanismo de Atuação.....	30
3.2.1.4 Funcionamento Básico da Catraca .....	31
3.2.2 Aplicação de Cadastro .....	33
3.2.2.1 Novo Cadastro .....	34
3.2.2.2 Editar Cadastro .....	38
3.2.2.3 Excluir Cadastro .....	39
3.2.2.4 Validar Entrada.....	40
3.2.2.5 Limpar Acessos Para um Novo Jogo.....	43
3.2.3 Servidor das Catracas.....	44
3.2.3.1 Interface com as Catracas .....	44
3.2.3.1 Interface com o Administrador do Sistema .....	49
3.2.4 Servidor de Banco de Dados .....	50
<b>4 RESULTADOS OBTIDOS .....</b>	<b>52</b>
<b>4.2 TEMPO DE LIBERAÇÃO .....</b>	<b>52</b>
<b>4.3 CONFIABILIDADE.....</b>	<b>53</b>
<b>4.3 CUSTO.....</b>	<b>53</b>

<b>5. PLANO DE NEGOCIOS.....</b>	<b>54</b>
<b>5.1 DEFINIÇÃO DO PRODUTO .....</b>	<b>54</b>
5.1.1 Natureza do Negócio.....	54
5.1.2 Missão .....	55
5.1.3 Valores .....	55
5.1.4 Visão.....	55
5.1.5 Objetivos.....	55
<b>5.2 DEFINIÇÃO DO PRODUTO .....</b>	<b>55</b>
5.2.1 Descrição.....	55
5.2.2 Tecnologia e Descrição Técnica .....	56
5.2.3 Vantagens Competitivas .....	57
<b>5.3 DEFINIÇÃO DO MERCADO.....</b>	<b>57</b>
5.3.1 Segmentação do Mercado .....	57
5.3.2 Segmento Alvo .....	57
5.3.3 Tamanho .....	58
5.3.4 Necessidade.....	58
5.3.5 Crescimento .....	58
5.3.6 Cliente Alvo .....	58
5.3.7 Aplicação .....	58
<b>5.4 ESTRATÉGIA DE MARKETING.....</b>	<b>59</b>
5.4.1 Preço .....	59
5.4.2 Publicidade .....	59
5.4.3 Promoção .....	59
5.4.4 Instalação .....	59
<b>5.5 ESTRUTURA ORGANIZACIONAL .....</b>	<b>59</b>
5.5.1 Estrutura Legal .....	59
5.5.2 Infraestrutura .....	60
5.5.2.1 Produção .....	60
5.5.2.2 Fornecedores .....	60
5.5.2.3 Logística .....	60
<b>5.6 PLANO FINANCEIRO.....</b>	<b>61</b>
5.6.1 Contabilização .....	61
5.6.2 Custo .....	62
5.6.3 Fluxo de Caixa Esperado .....	63
<b>6 CONCLUSÃO .....</b>	<b>65</b>
<b>6.1 CONSIDERAÇÕES SOBRE O DESENVOLVIMENTO DO PROJETO.....</b>	<b>65</b>
<b>6.2 OBJETIVOS ALCANÇADOS .....</b>	<b>66</b>
<b>REFERÊNCIAS.....</b>	<b>68</b>

## 1 INTRODUÇÃO

### 1.1 CONTEXTUALIZAÇÃO

Considerado um dos esportes mais populares do mundo, o futebol agrega muitos adeptos, sejam eles torcedores, praticantes ou profissionais.

No ano de 2012, no campeonato brasileiro da série A, foi registrada uma média de público de 12.983 torcedores por jogo (CONFEDERAÇÃO BRASILEIRA DE FUTEBOL, 2013). Esse alto número reflete uma necessidade constante de melhoria nas condições de acomodação e segurança dos estádios.

Atualmente, os sistemas eletrônicos de controle de acesso estão presentes nos mais variados locais, inclusive em estádios de futebol. Nestes, a maioria dos sistemas de controle de acesso funciona a partir da venda de ingressos físicos no formato de um cartão.

### 1.2 MOTIVAÇÃO E JUSTIFICATIVA DO PROJETO

O último tumulto significativo dentro de um estádio brasileiro ocorreu em dezembro de 2009, no estádio Major Antônio Couto Pereira. Nessa ocasião, vários torcedores perturbaram a ordem nas arquibancadas, invadiram o gramado e entraram em confronto com a polícia, cujo resultado foi um prejuízo de mais de R\$1.5 milhões em patrimônio depredado e mais de 20 feridos (ALVES, 2013).

Outro fato lamentável ocorreu com a morte de Kevin Douglas Beltrán Espada, durante uma partida pela Copa Libertadores da América, no estádio Jesús Bermúdez, na cidade de Oruro, na Bolívia. O adolescente foi atingido no olho por um sinalizador disparado pela torcida adversária, em jogo ocorrido dia 20 de fevereiro de 2013 (RIBEIRO, 2013).

Esses e outros casos mostram a necessidade da criação de medidas eficazes para a identificação e, em caso de necessidade, punição de torcedores.

Inúmeras são as medidas que tomadas por órgãos públicos para o aumento na segurança nos estádios, como o Governo Federal, Ministério Público e os governos estaduais para responsabilização civil de torcedores que promoverem desordens nos arredores e nos estádios.

“Torcida Legal” é um projeto já existente e trata de um conjunto de medidas que buscam a melhoria das condições de segurança e conforto nos estádios brasileiros.

Uma dessas iniciativas é a assinatura de um Termo de Cooperação entre o Ministério do Esporte, Ministério da Justiça, Conselho Nacional de Justiça, Conselho Nacional dos Procuradores Gerais e Confederação Brasileira de Futebol, que permite a prática de algumas medidas integradas e coordenadas de aperfeiçoamento das condições de segurança dos estádios (MINISTÉRIO DO ESPORTE, 2010).

Dentro desse conjunto de ações está também a assinatura do Decreto 6795/09 e a aprovação da Lei nº 12.299/10, que alterou o Estatuto do Torcedor ao regulamentar o controle das condições de segurança dos estádios desportivos e criar normas e padrões para o funcionamento das arenas em nosso país.

Os Estados e as torcidas têm no Ministério do Esporte o interveniente para fortalecer outra ação prevista no Torcida Legal e exigida pelo novo Estatuto do Torcedor: estabelecer o cadastramento dos torcedores organizados.

A primeira etapa do cadastramento ocorre no site [www.torcidalegal.gov.br](http://www.torcidalegal.gov.br), onde os torcedores organizados fornecem seus dados básicos. Após isso, o torcedor deverá procurar um local de atendimento para apresentação de documentos e coleta de foto e dados biométricos (MINISTÉRIO DO ESPORTE, 2010).

Com o Protótipo de sistema de Catracas Biométricas para Estádios de Futebol (PSCBEF), é possível o cadastro das impressões digitais e das fotos dos torcedores, como o projeto “Torcida Legal” já prevê, com algumas vantagens.

A primeira é que todos os torcedores, não somente os participantes de torcidas organizadas, terão acesso às dependências do estádio e, invariavelmente, precisarão do cadastro. Nesse cadastro haverá tanto dados identificadores como o nome, endereço, RG e CPF como a impressão digital do torcedor e também uma foto. Assim, há um controle mais eficaz de todas as pessoas que adentram ao estádio.

Outra vantagem é a eliminação, por completo, da ação de cambistas. Como o acesso será unicamente através da autenticação do torcedor por meio da impressão digital, não haverá ingresso físico e por consequência dispensará cambistas.

A praticidade é outro atrativo do PSCBEF. Sem o ingresso físico, haverá maior agilidade e praticidade na aquisição da entrada e no momento do acesso ao estádio.

## **2 FUNDAMENTAÇÃO TEÓRICA**

### **2.1 BIOMETRIA**

Biometria [bio (vida) + metria (medida)] é o estudo de características fisiológicas ou comportamentais únicas dos seres vivos. O termo também é comumente associado, em seres humanos, a uma identificação pessoal única, obtida a partir da análise dessas características (CANEDO, 2011).

Os sistemas biométricos funcionam a partir de características presentes em diversas partes do corpo humano, como por exemplo, a retina ou a íris, a palma da mão, as impressões digitais, as características do rosto, a voz, entre outros. O princípio no qual esses sistemas se fundamentam é na individualidade na unicidade do indivíduo que possui características físicas distintas.

#### **2.1.1 Impressões Digitais**

No corpo humano existem as chamadas papilas dérmicas, que são pequenas extensões da derme sobre a epiderme. A função dessas papilas é o aumento do contato entre a derme e a epiderme, para promover maior irrigação e enervamento da superfície da pele.

Na superfície dos pés e das mãos, essas papilas dérmicas aparecem como cristas papilares ou epidérmicas, e são comumente chamadas de impressões digitais (HARRIS, 2011).

O padrão criado por essas papilas é geneticamente determinado antes do nascimento. Esses permanecem inalterados, exceto em tamanho, durante toda a vida e determinam os padrões de impressões digitais, úteis em funções de identificação pessoal. Ou seja, as impressões digitais são uma marca exclusiva de cada indivíduo, mesmo entre gêmeos idênticos.

Impressões digitais podem ser deixadas em qualquer superfície tocada. No momento em que há o contato da pele com a superfície, as secreções naturais

presentes findam por aderir ao material e formam um padrão único devido às reentrâncias e saliências da pele (HARRIS, 2011).

Existem vários métodos de obtenção de impressões digitais, desde o mais simples, a partir do contato da mão com tinta na superfície de um papel e sua posterior análise, até os mais sofisticados com uso de sensores óticos e capacitivos.

#### 2.1.1.1 Sensores de Impressão Digital

Os tipos mais comuns de sensores de impressão digital são dois: o capacitivo e o puramente ótico. O princípio de funcionamento de ambos é baseado na obtenção de uma imagem da área da pele a ser analisada.

O processo de obtenção das digitais em sensores puramente óticos tem início quando o usuário coloca seu dedo sobre uma base de vidro e uma câmera registra a foto. Esta utiliza o chamado dispositivo de carga acoplado, também utilizado em câmeras fotográficas digitais. Este dispositivo nada mais é que um grupo de diodos fotossensíveis chamados *photosites*, que emitem um sinal elétrico na presença de fótons luz. O sinal emitido é processado e a imagem é criada (HARRIS, 2011).

Já nos sensores capacitivos, a maneira como a impressão digital é captada é diferente. A imagem não é obtida através da luz, e sim da corrente elétrica gerada por um grande número de pequenos capacitores, como descrito a seguir.

No lugar da base de vidro presente no sensor ótico, é utilizado um material semicondutor que contém minúsculas células, menores do que a largura de um sulco do dedo. Cada célula contém duas placas condutoras cobertas com uma camada isolante.

No momento em que o dedo é posicionado, nas células onde as pequenas saliências da pele se encostam às duas placas, a capacitância mostra-se diferente daquelas células onde não há contato das placas com a superfície do dedo. A partir desse princípio, e de uma série de amplificadores, é possível obter um sinal de corrente elétrica que contém a imagem da impressão digital (HARRIS, 2011).

### 2.1.1.2 Identificação Através da Impressão Digital

A identificação através da digital, também conhecida como papiloscopia, é o processo de comparação entre dois padrões de desenho criados pelas papilas dérmicas presentes na superfície da pele.

No intuito de reduzir os esforços no processo de identificação, as diferentes impressões digitais são classificadas quanto ao seu tipo. As pesquisas subsequentes servindo-se de métodos que utilizam características granulares da impressão, como o método que emprega as chamadas minúcias, ou *minutiae*.

Existem diversos sistemas de classificação de impressões digitais, entre os mais populares figuram: o sistema Roscher, o sistema Juan Vucetich e o sistema Henry de classificação.

Tomando como exemplo o sistema Henry, que é o mais utilizado por especialistas dentre os três, as impressões digitais são basicamente classificadas em três tipos: “*loop*”, “*whorl*” e “*arch*”, mostrados na figura 1, e que representam respectivamente 60-65%, 30-35% e 1-5% de todas as impressões digitais. Ainda há o chamado padrão “*composite*”, onde há a combinação dos três padrões. Há também classificações mais complexas as quais subdividem os padrões, como por exemplo: “*plain arches*” ou “*tented arches*”, que especificam ainda mais (HENRY, 1900, p.17).

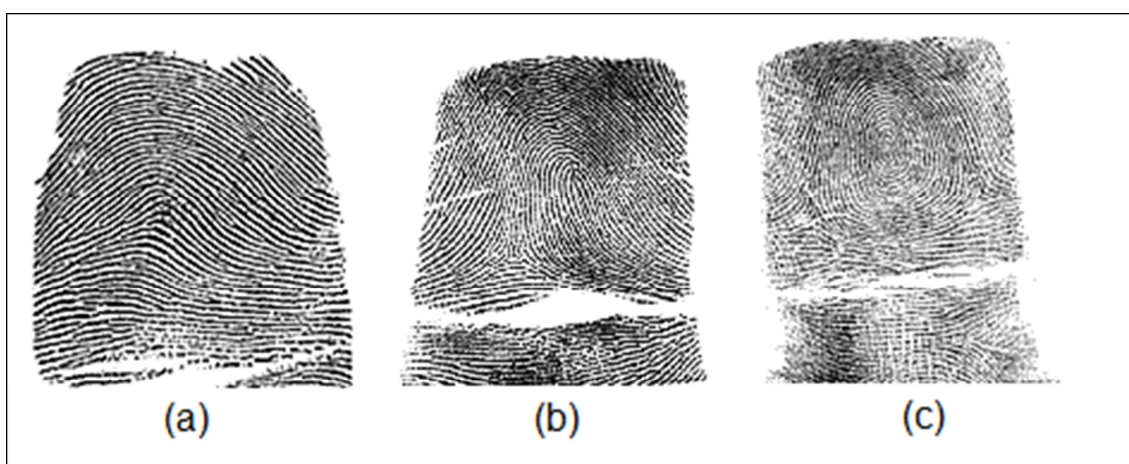


Figura 1 - Impressões digitais do tipo “arch” em (a), “loop” em (b) e “whorl” em (c).

Fonte: National Institute of Standards and Technology (2005).



Nesse sistema cada dedo recebe um valor numérico de acordo com a ordem na qual é localizado na mão, iniciado pelo dedo polegar direito como número um e finaliza no dedo mínimo esquerdo como número 10. É também designado um valor numérico aos dedos que contêm um padrão “*whorl*”. Neste caso, os dedos um e dois recebem valor 16, os dedos três e quatro, valor oito e, assim sucessivamente, até que os últimos dois dedos tenham valor um. Dedos com um padrão que não seja o “*whorl*” recebem valor nulo. A soma dos valores atribuídos aos dedos pares é colocada no denominador de uma função, e a dos dedos ímpares, no numerador. É somado, então, o valor de um no numerador e no denominador. Assim, a classificação da impressão digital é uma fração entre 1/1, que representa ausência de padrões “*whorl*” e 32/32, que representa existência desse padrão em todos os dedos (HENRY, 1900, p.73).

Uma vez classificada, a comparação entre impressões torna-se menos onerosa. A classificação, porém, não garante a identificação única. Para tal, são utilizados métodos de análise mais profunda dos padrões presentes nos dedos.

Os métodos de comparação mais detalhados podem ser divididos, basicamente, em dois: o método das minúcias e o método de correlação.

#### 2.1.1.2.1 Método das Minúcias

O método das minúcias, ou, em inglês, “*minutiae*”, mostram pequenas características únicas das cristas dérmicas em cada impressão digital. Estas características são constituídas de pequenos padrões, com ângulos distintos em áreas específicas da impressão digital. Tipicamente são do tipo “*ridge ending*” ou final de crista, a qual se apresenta como um fim abrupto em uma crista; “*bifurcation*” ou bifurcação, cujas cristas se dividem em duas; e os chamados “*short ridge*”, ou em uma tradução literal, crista curta, que são pequenas cristas significativamente menores que as outras cristas presentes (MAZUMDAR; VENKATA, 2008, p.3). Todos esses padrões são mostrados na figura 2.

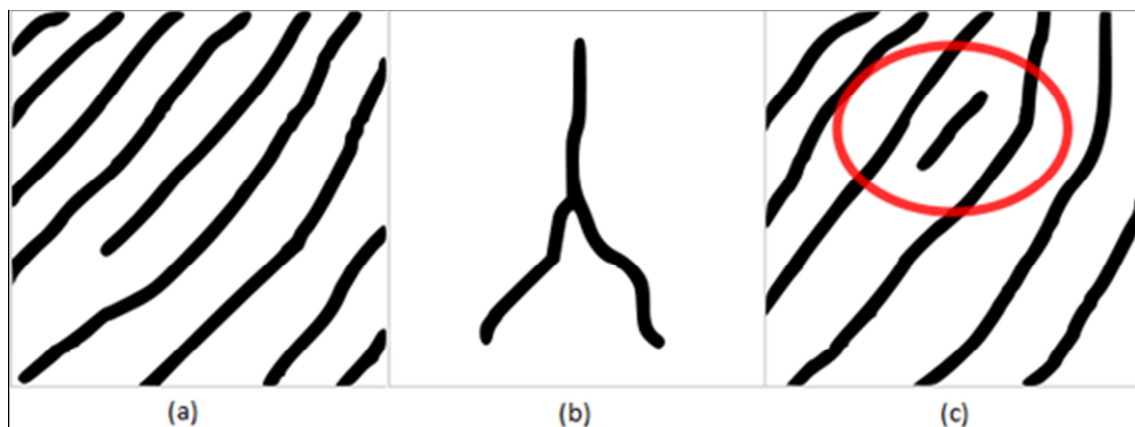


Figura 2 – Minúcias do tipo: “ridge ending” em (a), “bifurcation” em (b) e “short ridge” em (c).

Fonte: Mazumdar, Dhulipala (2005).

As técnicas baseadas no método das minúcias consistem na extração dos pontos que apresentam essas características e no posterior mapeamento e comparação dessas. Porém, há alguns problemas nessa abordagem: é bastante oneroso extrair esses pontos específicos quando a imagem possui baixa qualidade, além de não levar em conta o padrão global de cristas e sulcos.

#### 2.1.1.2.2 Método da Correlação

O método da correlação não leva em conta as características específicas ou minúcias da impressão digital. São consideradas todas as informações relativas às cristas das impressões digitais, não apenas seções específicas.

Como mostrado na figura 3, a cada crista são associados pequenos vetores, e estes, quando comparados entre imagens, fornecem um índice de coerência entre as duas imagens.

No entanto, existem alguns pontos negativos nessa abordagem. O método da correlação requer a precisa localização do ponto central da imagem, bem como seus ângulos de rotação e translação, o que pode torná-lo esse método consideravelmente complexo (PRABHAKAR; JAIN, 2007).



**Figura 3 - Exemplo de associação de vetores as impressões digitais.**

**Fonte: Biometrics Research Group – Michigan State University (2007).**

#### 2.1.1.2.3 Confiabilidade

Devido a variáveis externas como posicionamento, deformações, ruídos, umidade no dedo e condições do sensor, é impossível que duas imagens, obtidas em momentos distintos, coincidam exatamente. Por essa razão, a identificação deve ser feita através de um algoritmo que calcula o nível global de similaridade das impressões digitais.

O desempenho de um sistema de identificação de impressões digitais pode ser mostrado na figura 4, onde são evidenciadas todas as probabilidades de ocorrência no momento da identificação através da digital.

	$\omega_0$	$\omega_1$
$\hat{\omega}_0$	<i>TRR</i>	<i>FRR</i>
$\hat{\omega}_1$	<i>FAR</i>	<i>TAR</i>

**Figura 4 - Possibilidades de performance.**

**Fonte: Bazen et al.(2000).**

Nessa matriz,  $\omega_0$  é o usuário não cadastrado e  $\omega_1$  o cadastrado, com suas respectivas classes. Assim, se o usuário não cadastrado é identificado como tal, temos a TRR (*True Rejection Rate* – Taxa de verdadeira rejeição), porém, se é identificado como cadastrado, temos a FAR (*False Acceptance Rate* – Taxa de falsa

aceitação). Já para o usuário cadastrado,  $\omega_1$ , se houver identificação negativa, a probabilidade é chamada de FRR (*False Rejection Rate* – Taxa de falsa rejeição), caso haja a identificação desse usuário cadastrado como ele mesmo temos a TAR (*True Acceptance Rate* – Taxa de verdadeira aceitação) (BAZEN et al, 2000).

Para a maioria dos sistemas de identificação por impressões digitais, essas probabilidades podem ser controladas por meio de um parâmetro que define um limiar para similaridade entre as imagens.

## 2.2 BANCO DE DADOS

Bancos de dados (BD) são coleções de informações relacionadas a um determinado assunto e em uma determinada ordem. São essenciais em organizações e, com a maior informatização dos processos, há duas décadas ganharam extrema importância e se tornaram peça fundamental dos sistemas de informação.

### 2.2.1 Sistemas Gerenciadores de Bancos de Dados

Qualquer organização, por menor que seja, armazena milhares de dados cuja manipulação manual, além de ser mais demorada, pode resultar em erros devido ao desgaste do operador. É com a finalidade de amenizar esses problemas que surgiram os Sistemas Gerenciadores de Banco de Dados (SGBD).

Um SGBD é um conjunto de software responsável por armazenar, modificar e extrair informação de uma base de dados. Também fornece métodos para manter a integridade dos dados guardados, manipular o fluxo desses dados e recuperar a informação em caso de falha do sistema, com segurança.

Tipicamente, um SGBD suporta linguagens do tipo *query*, as quais são de alto nível, projetadas para simplificar consideravelmente o desenvolvimento de aplicativos que façam essa interface com os dados.

### 2.2.1.1 Modelos de Sistemas Gerenciadores de Bancos de Dados

Os primeiros bancos de dados surgiram no fim da década de 60, em uma época na qual havia a necessidade de um sistema de informação que apresentasse flexibilidade. Basicamente, existem cinco modelos de SGBD, distinguidos mediante a apresentação dos dados que possuem (KIOSKEA.NET, 2013).

- **Modelo Hierárquico:** foi o primeiro concebido para SGBD. Nele, os dados são classificados de forma hierárquica, onde obedecem a uma árvore descendente, e utilizam ponteiros entre os diferentes registros.
- **Modelo Rede:** Assim como no modelo hierárquico são utilizados ponteiros entre os diferentes registros. Porém, cada registro filho pode ser ligado a mais de um registro pai a fim de criar conexões bastante complexas.
- **Modelo Relacional:** É tido como o primeiro modelo de dados descrito teoricamente e se baseia na teoria matemática dos conjuntos. De uma maneira simples, é possível basear o modelo em três conceitos: de entidade, atributos e relações. Uma entidade, também chamada de tabela, é um elemento definido por todos os dados presentes. Cada entidade possui campos próprios, designados para informações específicas chamados de atributos. A relação determina o modo como cada registro de cada tabela se associa a registros de outras tabelas.

Atualmente é o modelo mais utilizado, e surge em cerca de 80% dos bancos de dados existentes.

- **Modelo Dedutivo:** Assim como o modelo relacional, os dados ficam em tabelas, porém, estas são manipuladas através de cálculos de predicados.
- **Modelo Objeto:** Os princípios desse modelo são os mesmos da programação orientada a objetos. Utiliza classes com atributos e os campos a serem armazenados são instâncias dessas classes (KIOSKEA.NET, 2013).

### 2.2.2 Linguagem SQL

A *Structured Query Language* ou SQL, baseada na álgebra relacional, é a linguagem padrão para bancos de dados relacionais.

A SQL foi desenvolvida no começo da década de 70, nos laboratórios da IBM em San Jose. Pertencia ao projeto System R, que tinha por objetivo demonstrar a viabilidade da implementação do modelo relacional proposto anteriormente.

Devido 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 na linguagem SQL são divididas em, basicamente, quatro conjuntos:

- 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 pertencentes ao conjunto DML permitem ao usuário a manipulação dos dados em uma tabela. Inclusões, consultas, alterações são efetuadas através destas. São comandos típicos desse conjunto: *SELECT*, *INSERT*, *UPDATE*, *DELETE*, *MERGE*, *CALL*, *EXPLAIN PLAN* e *LOCK TABLE* (ORAQFAQ, 2004).

Já o conjunto DDL tem como função a definição do esquema, ou estrutura da base de dados. São comandos típicos desse conjunto: *CREATE*, *ALTER*, *DROP*, *TRUNCATE*, *COMMENT*, e *RENAME*.

O DCL assume o controle de autorização de dados e licenças de usuários. Esse conjunto contém tipicamente as instruções *GRANT* e *REVOKE*.

O conjunto DTL gerencia as mudanças feitas pelas instruções do grupo DML cujo grupo permite que instruções sejam agrupadas em padrões lógicos. Instruções típicas: *COMMIT*, *SAVEPOINT*, *ROLLBACK*, *SET TRANSACTION* (ORAQFAQ, 2004).

### 2.3 Plataforma .NET

A plataforma *Microsoft .NET* é um projeto da empresa Microsoft com o intuito de ser uma plataforma 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, 2012).

### 2.3.1 Linguagem C#

A linguagem de programação C# é baseada no C++, emprestando características do *Object Pascal* e do *Java*. Define-se, basicamente, pela orientação a objetos e forte tipagem. Foi desenvolvida pela *Microsoft* como parte da plataforma .NET (MSDN, 2012).

A linguagem permite o uso de ponteiros, porém, o seu uso não é recomendado devido à necessidade de uma requisição de permissões mais alta. Existem, também, nas operações aritméticas, métodos de controle que evitam o transbordamento de dados. Um coletor de lixo, processo usado para a manutenção de memória, está também disponível. Com este último, é possível recuperar uma área da memória que o aplicativo já deixou de usar. A herança múltipla não é suportada (MSDN, 2012).

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, 2012).

## 3 DESENVOLVIMENTO DO PROJETO

O projeto em questão consiste em uma solução completa de controle de acesso via impressões digitais, em princípio idealizados para uso em estádios de futebol.

A figura 5 representa os principais componentes da solução em questão. Foi utilizado como controladora da catraca uma placa *LPCXpresso-1769* com a *LPCXpresso Baseboard* do fabricante *Embedded Artists*. O servidor de banco de dados foi desenvolvido na aplicação *SQL Server 2012* da *Microsoft*. O servidor das catracas e a aplicação de cadastro, foram desenvolvidos em linguagem *C#*, mais precisamente na plataforma *.NET* também da *Microsoft*.

A maior parte das comunicações entre os componentes do sistema é feita via Ethernet, com TCP/IP. A catraca possui um sensor biométrico RS232 que se comunica diretamente com a controladora. Para controle do mecanismo, é usado um driver para acionamento do solenoide, controlado diretamente pelos pinos do processador.

### 3.1 FUNCIONAMENTO BÁSICO

Inicialmente há o cadastro e validação dos torcedores em uma aplicação específica desenvolvida em *C#*. Essas informações geradas durante o cadastro são transmitidas ao servidor de banco de dados, via TCP/IP, e armazenadas.

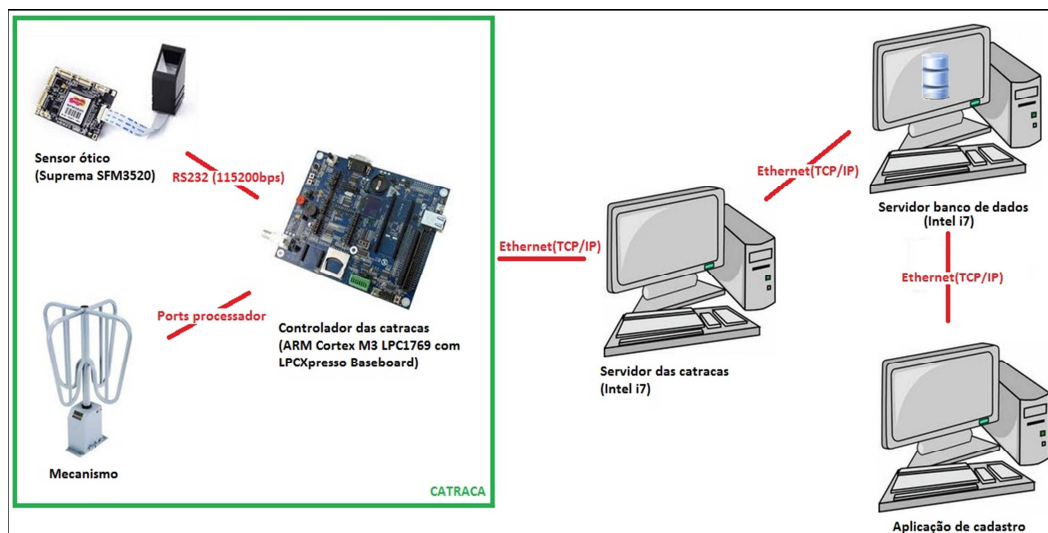


Figura 5 – Representação do Sistema.

Fonte: Autoria própria.



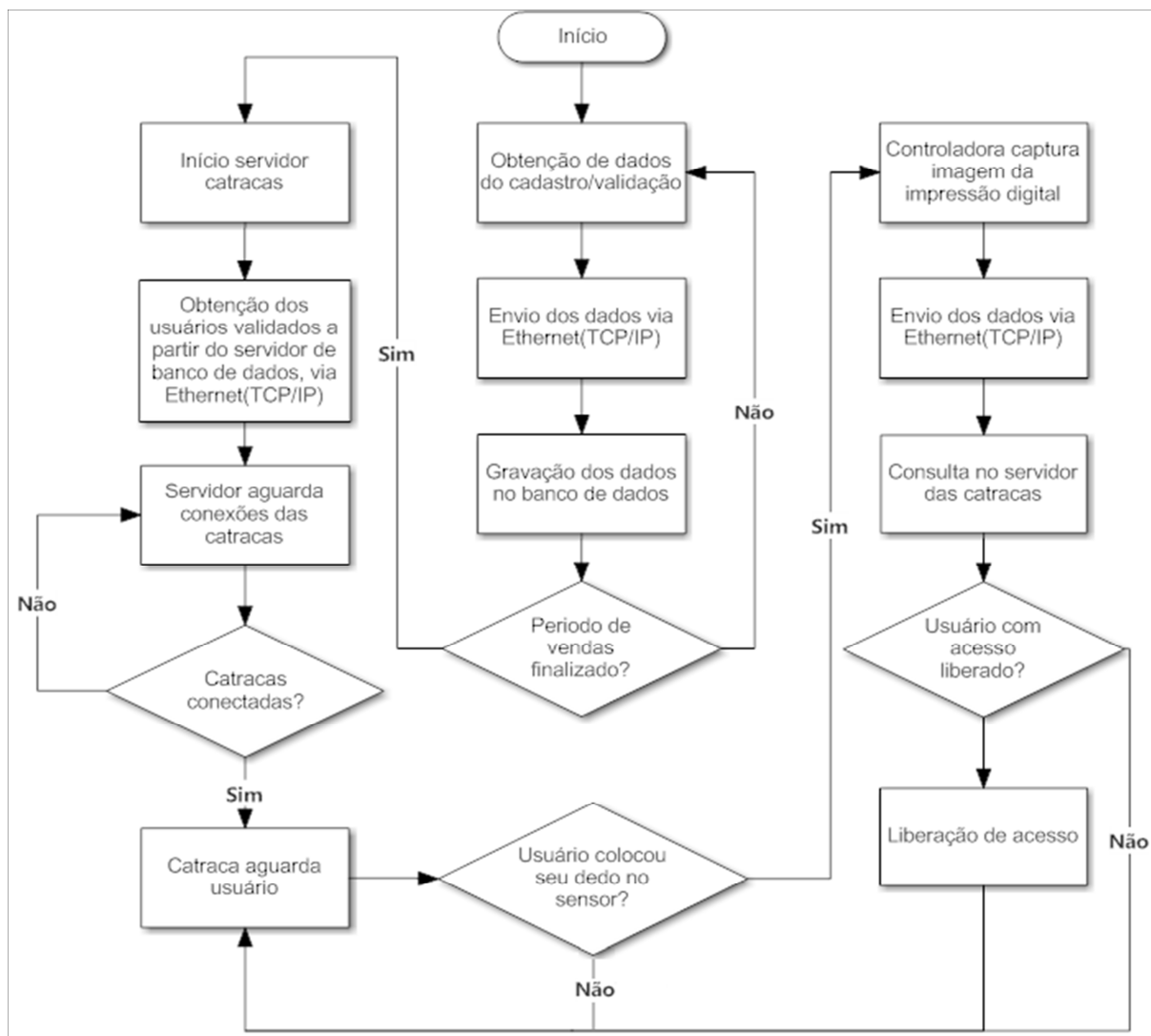


Figura 6 - Fluxograma de funcionamento.

Fonte: Autoria própria.

Após o término da venda dos ingressos é necessário iniciar o servidor das catracas, também desenvolvido em C#, que se conecta ao servidor de banco de dados, via TCP/IP, para obter todas as digitais dos cadastros válidos para acesso. Após este processo, o servidor fica no aguardo de conexões com as catracas. A controladora da catraca, então, se conecta ao servidor e permanece à disposição do usuário. Ao detectar uma imagem vinda do sensor, a controladora envia esses dados ao servidor, que, por sua vez, procura em um banco de dados próprio no qual estão as impressões cadastradas para aquele evento e envia uma resposta à controladora da catraca. A partir dessa resposta, a controladora efetua ou não a

liberação do mecanismo de acesso. A figura 6 mostra um fluxograma de funcionamento do sistema desenvolvido.

## 3.2 COMPONENTES DO SISTEMA

### 3.2.1 Catracas

Cada catraca é composta por três componentes: uma placa controladora, o sensor de impressões digitais e um mecanismo de atuação.

#### 3.2.1.1 Placa Controladora

Para o desenvolvimento desse projeto foi utilizado o kit *LPCXpresso*, mostrado na figura 7, composto pela placa *LPCXpresso Baseboard* em conjunto com a placa *LPCXpresso 1769*.

O *LPCXpresso* é uma plataforma de baixo custo desenvolvida com microcontroladores da família ARM fabricados pela NXP e trabalha em conjunto com um IDE de nome *LPCXpresso 4*, criado pela *CodeRed*.

O módulo de desenvolvimento *LPCXpresso 1769* conta com um microcontrolador *Cortex M3*, da família ARM, e uma interface JTAG para auxiliar o processo de desenvolvimento e *debug* de *firmware*.

A *LPCXpresso Baseboard* é uma facilitadora de prototipagem, oferece um conjunto de periféricos que auxiliam e facilitam o desenvolvimento do projeto. Entre os periféricos a placa oferece acelerômetro, display, sensor de temperatura, ponte SPI – UART e ponte USB para serial.

O acionamento do solenoide e dos LEDs indicativos é feito de maneira bastante simples a partir da polarização de transistores. Já a interface serial utiliza o circuito integrado MAX232, que recebe, normaliza e envia ao sensor biométrico os dados provenientes do processador. Uma placa de interface foi construída para a comunicação com o módulo biométrico e acionamento do solenoide e LEDs de indicação. O diagrama desta placa é mostrado na figura 8.

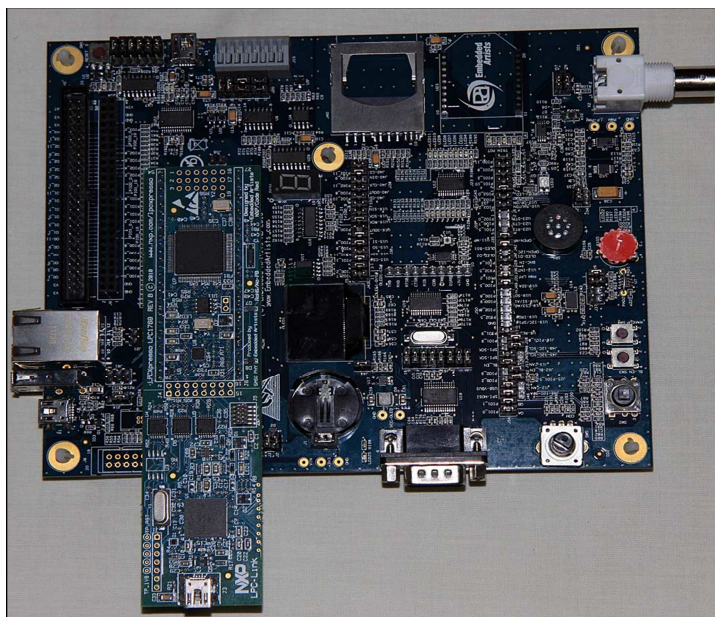


Figura 7 - LPCXPRESSO BaseBoard.

Fonte: Autoria própria.

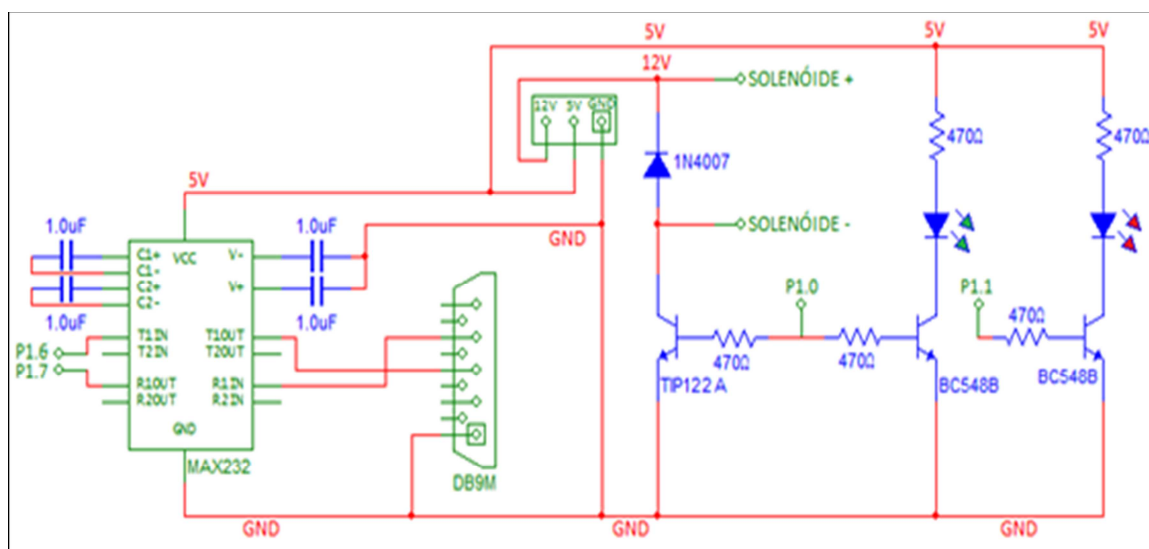


Figura 8 - Diagrama esquemático da placa de interface.

Fonte: Autoria própria.

O acionamento do solenoide e dos LEDs indicativos é feito de maneira bastante simples a partir da polarização de transistores. Já a interface serial utiliza o circuito integrado MAX232, que recebe, normaliza e envia ao sensor os dados provenientes do processador. Uma placa de interface foi construída para a

comunicação com o módulo biométrico e acionamento do solenoide e LEDs de indicação. O diagrama desta placa é mostrado na figura 8.

### 3.2.1.2 Sensor Biométrico

Tanto na catraca quanto na aplicação de cadastro, é utilizado o sensor biométrico SFM3520 do fabricante Suprema, mostrado na figura 9.



Figura 9 - Sensor biométrico SFM3520 do fabricante Suprema.

Fonte: Suprema.

Esse sensor biométrico é utiliza recursos DSP, desenvolvido dentro dos conceitos de desempenho com baixo consumo. Obtém a imagem da impressão digital de maneira semelhante a uma câmera fotográfica digital através de um dispositivo óptico de carga acoplado.

O sensor tem a capacidade de gerar três modos de imagem distintos: imagem do tipo binária, utilizada no projeto, imagem do tipo escala de cinza e em escala de cinza no padrão 4bits. Essas imagens geradas possuem resolução de 500 dpi e tamanho de 272x320 pixels (SUPREMA, 2012).

A alimentação do sensor é feita através de uma fonte 5V e seu consumo de corrente é tipicamente entre 140 e 175mA. A comunicação é feita primordialmente a partir de uma interface RS232, com taxa de transmissão de 115kbps (SUPREMA, 2012).

Conta com uma memória interna e possui funcionalidade própria de cadastro e verificação de impressões digitais, porém, devido aos requisitos deste projeto,

essa funcionalidade não é utilizada. A função *Scan Image*, que obtém a imagem da impressão digital e envia através da porta serial, é utilizada neste projeto.

O protocolo de comunicação do sensor é bastante simples. A comunicação é feita através de pacotes padronizados, ou seja, cada evento do sensor possui um respectivo pacote. Estes pacotes são formados por 13 bytes, e sua estrutura é mostrada na figura 10.

	Início do pacote	Comando	Parâmetro	Tamanho	Flag/Erro	Verificador	Fim de pacote
Tamanho (bytes)	1	1	4	4	1	1	1
Valores comuns	Sempre 0x40	0x01(Confirmação), 0x15 (Scan Image)	Depende do comando	Depende do comando	0x61(Sucesso), 0x62(Sucesso no scan), 0x6C(Time out)	Soma de todos os bytes do pacote; se maior que 0xFF divide-se por 0x100 e utiliza-se o resto da divisão	Sempre 0x0A

Figura 10 - Estrutura do pacote de comunicação.

Fonte: Autoria própria.

A função *Scan Image* é enviada ao sensor biométrico, e o pacote é formado pelo código 0x40, 0x15, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x55 e 0x0A. Após o envio desse código, o sensor acende uma luz vermelha e aguarda pela impressão digital, se não houver nenhuma impressão dentro de 10 segundos ou se a imagem obtida for de baixa qualidade, o sensor responde com um pacote indicando erro e fica disponível para outro comando. Porém, se for adquirida uma impressão digital válida, o sensor responde um pacote com flag 0x62, então envia um novo pacote que contém a imagem. A estrutura desse pacote é mostrada na figura 11.

Apesar do campo tipo de compressão não ser utilizado (seu valor padrão é zero), os 10880 bytes representando a imagem possuem compressão onde cada bit representa um pixel.

	Largura	Comprimento	Tipo de compressão (não utilizado)	Encriptamento	Formato imagem	Tamanho da imagem	Imagem	Fim de pacote
Tamanho (bytes)	4	4	4	4	4	4	10880	1
Valores comuns	0x10; 0x01; 0x00; 0x00	0x40; 0x01; 0x00; 0x00	0x00; 0x00; 0x00; 0x00	0x00; 0x00; 0x00; 0x00	0x01; 0x00; 0x00; 0x00	0x00; 0x54; 0x01; 0x00	Varia	0x0A

**Figura 11 - Pacote da Imagem.**

**Fonte: Autoria própria.**

Em uma imagem bitmap de profundidade oito bits, cada pixel é representado por um byte, e o valor atribuído a esse byte é o que define a cor. E como há essa compressão característica, cada bit um será transformado em um byte com valor 255, que representa a cor preta, e cada bit zero será transformado em um byte com valor zero, que representa a cor branca (SUPREMA, 2012).

### 3.2.1.3 Mecanismo de Atuação



**Figura 12 - Catraca Minibloqueio da marca Wolpac.**

**Fonte: Wolpac.**

O mecanismo de liberação de acesso é baseado na catraca Minibloqueio do fabricante Wolpac, mostrada na figura 12.

Essa catraca possui um mecanismo com solenoide 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 solenoide quanto no de detectar quando o usuário passou completamente na catraca.

O solenoide requer uma tensão de 12V para ativação. Quando presente, essa tensão 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.

#### 3.2.1.4 Funcionamento Básico da Catraca

A primeira tarefa a ser realizada para o funcionamento da catraca (figura 13) é a configuração de IP. Cada controladora necessita de um IP único e conecta ao servidor por meio de uma porta distinta. Isto é feito a partir de um display e um pequeno *joystick* na placa controladora. O último octeto do IP recebe um valor maior quando o *joystick* é acionado para cima e um valor menor quando acionado para baixo. Para confirmar é necessário um clique no *joystick*.

Nas simulações, a faixa de IP utilizada para as catracas foi de 192.168.0.110 até 192.168.0.115. A catraca com o primeiro IP (192.168.0.110) conecta-se ao servidor através da porta 23, a segunda catraca utiliza a porta 24 e assim sucessivamente.

Após a configuração de IP, a controladora solicita conexão com o servidor, que possui IP 192.168.0.100.

Uma vez conectada ao servidor, a controladora envia o pacote com comando *Scan image* ao sensor biométrico e aguarda o primeiro pacote de resposta. Caso o primeiro pacote de resposta apresente a *flag* diferente de 0x62 (comando bem sucedido) os dados são descartados e o comando *Scan image* é enviado novamente. Já se a controladora receber a *flag* 0x62, é iniciado o processo de recebimento dos bytes que representam a imagem.

Esses são guardados em um *buffer*, e posteriormente enviados ao servidor sob o protocolo TCP/IP. Para isto é utilizada uma biblioteca em linguagem de programação C que implementa a pilha TCP/IP. Devido às limitações da placa

controladora, são necessários que o *buffer* seja dividido em três partes, que são enviadas separadamente, assim, os 10880 são divididos em três pacotes de 3627, 3627 e 3626 bytes, respectivamente, para serem enviados ao servidor.

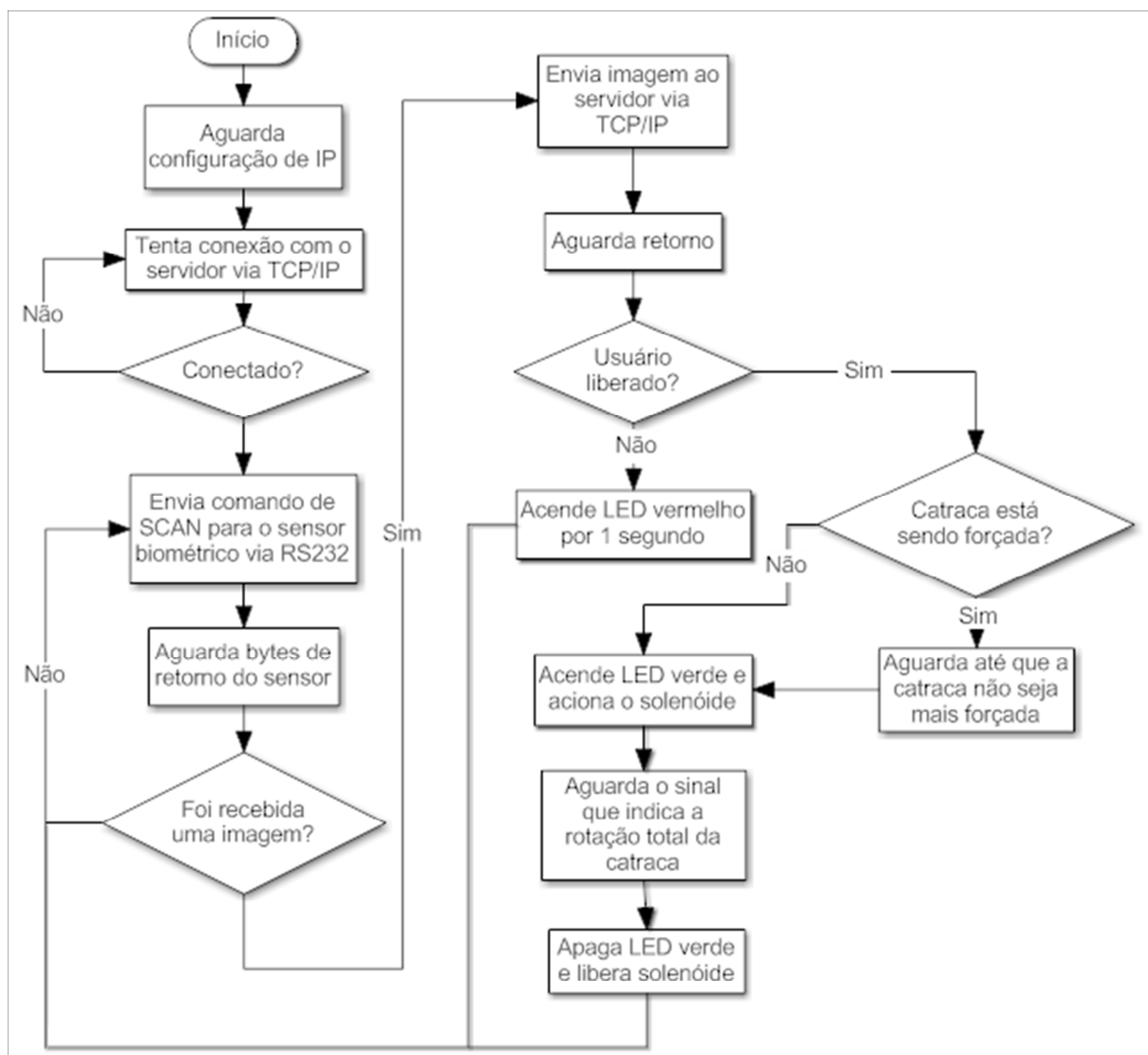


Figura 13 - Funcionamento da catraca.

Fonte: Autoria própria.

Enquanto o servidor processa os dados, a placa controladora aguarda a resposta que é bastante simples; se o usuário estiver liberado, o servidor manda um caractere “s” à controladora; se o usuário não possui acesso é enviado a controladora um caractere “n”.

Antes de executar a liberação ou não do torcedor, é necessário confirmar que a catraca não esteja sendo forçada, pois isso pode acarretar danos ao mecanismo.



Isto é feito a partir da leitura de um sensor, presente no mecanismo, que envia um sinal de 3.3V à controladora quando há esse tipo de movimento.

Quando a resposta é o caractere “n”, a controladora acende um LED indicativo na cor vermelha, por um período de um segundo e retorna ao estado de espera por uma nova impressão digital.

Já se o usuário possuir acesso, a placa controladora inicia o processo de liberação do usuário. Ao acionar o solenoide, o mecanismo é liberado para o giro das barras e, simultaneamente, a controladora acende um LED verde que indica a liberação de acesso.

O usuário, então, inicia o movimento de passagem pela catraca e, ao completar um giro completo, o sensor do mecanismo envia outro sinal de 3.3V ao processador, o LED verde é apagado, e o solenoide é desacionado para que a controladora entre em modo de espera por um novo usuário.

Todo o firmware da catraca foi desenvolvido na linguagem C, utilizando o IDE LPCXpresso 4.0, do fabricante *CodeRed*.

### 3.2.2 Aplicação de Cadastro

A aplicação de cadastro é um *software* em linguagem C# desenvolvido de maneira que seu uso fique o mais intuitivo possível e permita uma interface amigável ao usuário no momento de inserir os dados do torcedor no banco de dados. O funcionamento se dá através de janelas chamadas *forms*, botões para ações e caixas de texto chamadas *textbox* e é função dessa aplicação fazer a transmissão dos dados captados, ao banco de dados no servidor.

Quando o programa é aberto, a classe “Principal” instancia um novo objeto e abre uma janela (figura 14), que tem como função organizar seis botões: “Novo Torcedor”, “Validar Entrada”, “Editar Cadastro”, “Excluir Torcedor”, “Limpar acessos para um novo jogo” e “Sair”. Cada botão, com exceção do Sair, é relacionado com uma classe correspondente de mesmo nome, por exemplo, o botão “Excluir Torcedor” corresponde à classe Excluir. Quando um dos botões é acionado, a classe principal cria um novo objeto da classe correspondente. O botão “Sair” tem como única função fechar a aplicação.



Figura 14 - Tela Principal da Aplicação de Cadastro.

Fonte: Autoria própria.

### 3.2.2.1 Novo Cadastro

A screenshot of a Windows-style application window titled "Novo Cadastro de Torcedor". The window has a title bar with minimize, maximize, and close buttons. The main content area is light gray and features the title "Cadastro do novo Torcedor" in bold. Below the title are several input fields: "Nome:" (text box), "Endereço:" (text box), "Telefone:" (text box), "Data de Nasc.:" (text box), "R.G.:" (text box), "CPF:" (text box), and "Sexo:" with radio buttons for "Masculino" and "Feminino". To the right of the "Nome:" field is an "ID:" field containing the number "5". At the bottom, there are two large empty square boxes for photo capture, with buttons "Capturar Foto" and "Capturar Digital" below them. On the right side, there are two buttons: "Salvar e Sair" and "Voltar".

Figura 15 - Tela de Cadastro do novo Torcedor.

Fonte: Autoria própria.

A tela de novo cadastro, mostrada na figura 15, quando aberta, disponibiliza campos (*textbox*) para inserção dos dados referentes à pessoa, um botão para capturar a foto através da câmera presente no computador de cadastro, um botão para a captura da impressão digital através do módulo biométrico, um botão para salvar e sair, um botão para voltar e um número de identificação do torcedor (ID).

Na inicialização desta janela, o construtor desta classe faz uma consulta, através do comando “SELECT MAX(ID)” para se obter o número do último cadastro, o novo torcedor será então cadastrado no final da lista e o número de ID será único. O ID é utilizado sempre que é necessário pesquisar os dados de um torcedor já cadastrado no BD.

Os dados pessoais do torcedor são armazenados no banco de dados no formato de *string* e são inseridos através dos *textbox*, mostrados em destaque na figura 16. O uso desta estrutura implica na colocação manual de caracteres de separação como é o caso de pontos e traços nos campos Data de Nascimento, RG e CPF.

A escolha do gênero sexual do torcedor é feita por *RadioButton*, estrutura mostrada na figura 17 e permite que somente uma opção seja escolhida, e permanece armazenada em uma variável do tipo inteira (*int*). O valor um representa o sexo masculino e dois representa o sexo feminino. A opção do uso desse tipo de variável é pela possibilidade de uma eventual classificação dos Torcedores, já que em um ambiente real é possível destinar catracas específicas para homens e mulheres, com a separação do BD em dois bancos menores representando os sexos.



Nome:

Endereço:

**Figura 16 - Textbox.**

**Fonte: Autoria própria.**

Sexo:	<input type="radio"/> Masculino
	<input type="radio"/> Feminino

Figura 17 - *RadioButton*.

Fonte: Autoria própria.

A captura da foto da pessoa é realizada através da classe “WebCam” que é instanciada através do botão “Capturar Foto”. É aberta uma janela de nome igual ao da classe que possui um botão para iniciar o vídeo e capturar a foto e um segundo botão para voltar à tela anterior (figura 18). Essa classe aciona a câmera instalada no computador e através do botão iniciar é exibido o vídeo captado; quando a pessoa se posiciona adequadamente em frente à câmera, é feita a captura de um *frame* do vídeo e salvo em uma variável do tipo *image*. Quando se retorna à tela de cadastro é exibida a foto a ser salva no BD.

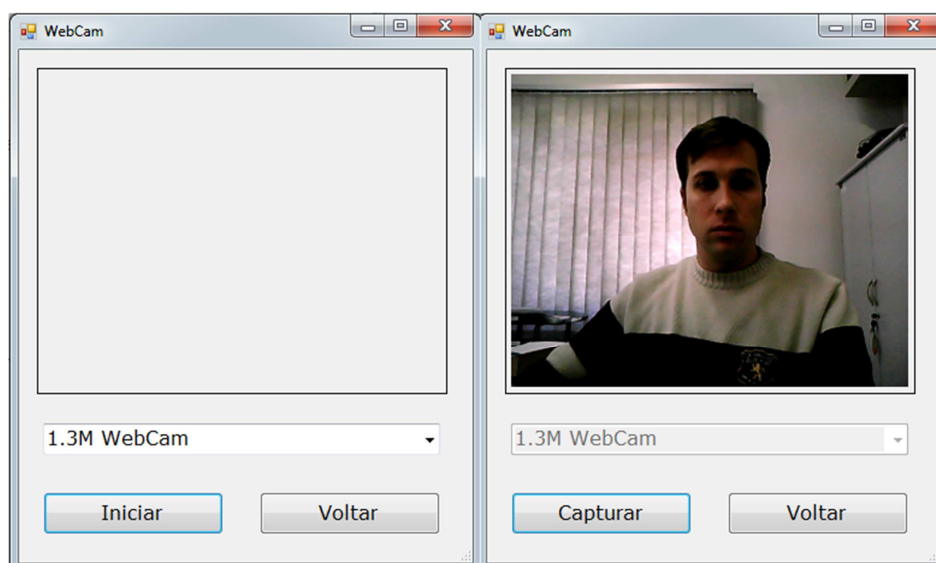


Figura 18 - Janela de captura de imagem.

Fonte: Autoria própria.

A classe responsável pela captura da impressão digital funciona de forma similar. Quando o objeto da classe “Digital” é instanciado, é verificado se o módulo biométrico está conectado à porta serial do computador de cadastro, para então abrir a janela para captura (figura 19). Então, da mesma forma como na classe

“WebCam”, é necessário apertar um botão para iniciar a captura da impressão digital.

Quando é acionado o botão de captura é aberta uma comunicação com a porta serial e enviado ao módulo um pacote de 13 bytes de dados com o comando de *ScanImage*. Assim que a pessoa coloca o dedo no sensor o módulo responde com um pacote de 10880 bytes de dados compactados de uma forma específica definida pelo fabricante. A classe então retira os primeiros 54 bytes que são de cabeçalho, e também o último byte que indica o final do pacote. O restante dos bytes devem ser convertidos para o formato *bitmap*, o que é feito pela função “CopyDataToBitmap”. Esta converte cada byte recebido em oito bytes. Suponha-se, por exemplo, o primeiro byte da imagem é o valor hexadecimal 0x42; a representação dos bits no formato binário é 0100 0010. A função converte os bits zeros em um byte 0x00 e os bits um em um byte 0xFF. No caso exemplificado, o valor 0x42 é convertido para a sequência 0x00 0xFF 0x00 0x00 0x00 0x00 0xFF 0x00. A imagem final terá, portanto, 88Kbytes. Ao voltar à tela de cadastro deve-se pressionar o botão “Mostrar” para que a imagem da impressão digital seja exibida.

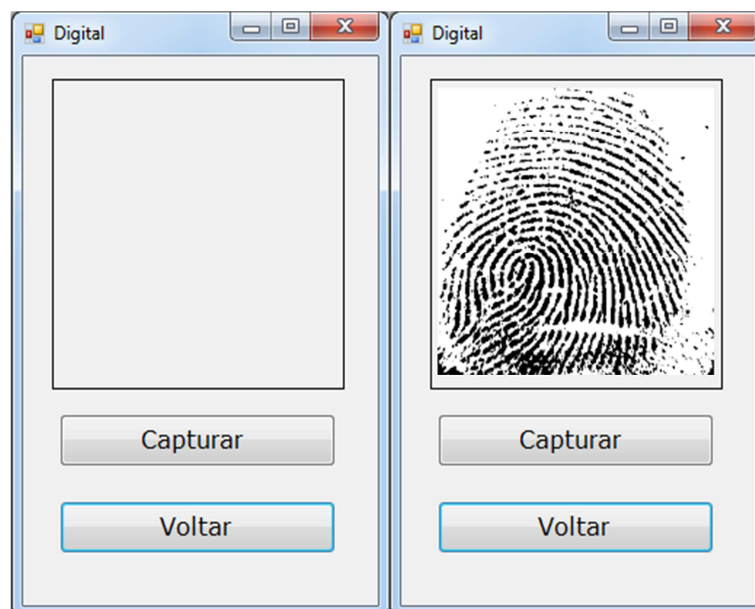


Figura 19 - Janela da captura da impressão digital.

Fonte: Autoria própria.

De volta à tela de cadastro, o botão “Salvar e Sair” tem como função enviar os dados ao BD, o que é feito através do comando INSERT INTO. Na figura 20 está o trecho do código SQL responsável pela inserção dos dados.

Os dados escritos nos *textbox* são passados diretamente ao BD, através da instrução `cmd.Parameters.AddWithValue("@NOME", tb_Nome.Text)`. A instrução `tb_Nome.Text` converte o texto do *textbox* em uma *string* de dados para ser inserida na coluna correspondente no BD.

A foto e a imagem da impressão digital do torcedor devem ser convertidas em um *byte array* que é o formato requerido pelo SQL.

O inserção, efetivamente, ocorre com a instrução `cmd.ExecuteNonQuery()`. Para que os dados sejam salvos corretamente todos os campos devem ser completados. Quando isso ocorre e a comunicação com o BD não apresenta problemas uma mensagem de conclusão é mostrada na tela.

```

84 private void button4_Click(object sender, EventArgs e)
85 {
86     SqlCommand cmd = new SqlCommand();
87     cmd.Connection = myConection;
88     cmd.CommandText = "INSERT INTO Torcedores (ID,Nome,Endereco,Telefone,RG,CPF,Nasc,Sexo,Foto,Digital,Acesso,Restri) +
89         "SELECT (@ID),(@NOME),(@END),(@TEL),(@RG),(@CPF),(@NASC),(@SEX),(@FOTO),(@DIGI),(@ACESSO),(@RESTRI)";
90     cmd.Parameters.AddWithValue("@ID", l_ID.Text);
91     cmd.Parameters.AddWithValue("@NOME", tb_Nome.Text);
92     cmd.Parameters.AddWithValue("@END", tb_End.Text);
93     cmd.Parameters.AddWithValue("@TEL", tb_fone.Text);
94     cmd.Parameters.AddWithValue("@RG", tb_RG.Text);
95     cmd.Parameters.AddWithValue("@CPF", tb_CPF.Text);
96     cmd.Parameters.AddWithValue("@NASC", tb_Nasc.Text);
97     cmd.Parameters.AddWithValue("@SEX", sex);
98     cmd.Parameters.AddWithValue("@FOTO", foto_array);
99     cmd.Parameters.AddWithValue("@DIGI", Digital_array);
100    cmd.Parameters.AddWithValue("@ACESSO", 0);
101    cmd.Parameters.AddWithValue("@RESTRI", 0);
102    try
103    {
104        myConection.Open();
105        cmd.ExecuteNonQuery();
106        myConection.Close();
107        MessageBox.Show("Dados Salvos com Sucesso");
108    }
109    catch (Exception ex)
110    {
111        MessageBox.Show("Não foi possível salvar os dados! Verifique os dados novamente." +
112            " Todos os campos devem ser preenchidos" + ex);
113        myConection.Close();
114    }
115    Close();
116 }

```

Figura 20 - Trecho de código - Botão Salvar e Sair.

Fonte: Autoria própria.

### 3.2.2.2 Editar Cadastro

Esta tela funciona de forma semelhante à tela de cadastro, mas possui duas diferenças. A primeira é que o usuário indica o número de ID, e a partir deste

número os dados armazenados no servidor são carregados na tela (figura 21). Depois de carregados, qualquer dado pode ser alterado, com exceção do ID.

A segunda diferença para a tela de cadastro está na possibilidade de criar uma restrição para a validação da entrada do torcedor ao estádio. Ao selecionar a opção “Sim” considera-se de que o torcedor cometeu alguma falha que o clube julgue passível de punição, por exemplo, ficar afastado do estádio por um período de seis meses por ter atirado algum objeto dentro do campo. O espaço destinado ao texto de restrição tem capacidade de 150 caracteres e, sempre que a restrição do torcedor acabar, o texto deve ser deletado manualmente.

Quando se pressiona o botão “Salvar e Sair”, por uma opção de simplificar a programação, todos os dados do torcedor selecionado são sobrescritos. O funcionamento deste botão é idêntico ao da classe de cadastro, já descrita anteriormente.

A interface de usuário, intitulada "Edição dos dados do Torcedor", apresenta um formulário com os seguintes elementos:

- Um campo de busca com o botão "Buscar por ID".
- Campos de texto para "Nome:", "Endereço:", "Telefone:", "Data de Nasc.:", "R.G.:", e "CPF:".
- Opções de "Sexo:" com radio buttons para "Masculino" e "Feminino".
- Um checkbox para "Restrição Entrada do Torcedor" com a opção "Sim".
- Dois espaços reservados para imagens, cada um com um botão "Capturar Foto" e "Capturar Digital" respectivamente.
- Botões de ação "Salvar e Sair" e "Voltar".

Figura 21 - Edição do cadastro.

Fonte: Autoria própria.

### 3.2.2.3 Excluir Cadastro

Se por algum motivo for necessário excluir totalmente o cadastro de um torcedor, isso é feito através da classe “Excluir”. Quando a janela (figura 22) de

exclusão de cadastro é aberta deve-se inserir o número de ID do torcedor a ser excluído. Todos os dados desse torcedor são carregados na tela para conferi-los os e confirmar se é o torcedor a ser excluído.

Essa classe apaga totalmente os dados do torcedor selecionado sendo impossível a restauração dos dados; caso seja necessário, o torcedor deverá ser cadastrado novamente com um novo número de ID. Para tal tarefa é utilizado o comando *DELETE* do SQL. Supondo que o BD de determinado clube de futebol possua 100 torcedores cadastrados, quando for apagado qualquer registro entre o ID 2 e o ID 99, esse número de ID não será mais utilizado. Caso se apague o ID 100, o último ID, o próximo novo torcedor cadastrado obterá o registro 100.



Figura 22 - Excluir Torcedor.

Fonte: Autoria própria.

#### 3.2.2.4 Validar Entrada

A validação da entrada representa a venda do ingresso ao torcedor. Optou-se pelo termo validar porque a aplicação não tem nenhum recurso financeiro e sendo assim a validação consiste apenas na permissão do torcedor entrar ao estádio. No banco de dados existe um campo de nome "Acesso" que possui o valor zero caso o



torcedor não possua permissão de acesso e possui o valor um representando a permissão de entrada.

Na janela aberta (figura 23), a busca do torcedor ocorre através do número de ID e quando esses dados são encontrados no BD, eles apenas são exibidos na tela e não é possível editar nenhum campo através dessa função.

Na janela exibida existe um campo que representa o número de ingressos vendidos para determinado jogo. Considerando que esse sistema esteja instalado em um estádio de capacidade máxima definida, o *software* apenas emitirá a validação da entrada caso o número de ingressos vendidos seja inferior a esta capacidade. Para o desenvolvimento deste projeto, a capacidade do estádio está limitada a 10 torcedores. O cálculo da quantidade de ingressos vendidos é feita no momento da busca do torcedor: a classe “Val\_Ent” faz uma contagem de todos os torcedores que possuem o campo “Acesso” com valor um no BD e guarda em uma variável.

A imagem mostra uma janela de software intitulada "Validação da Entrada do Torcedor". No topo, há um campo de entrada para o ID do torcedor e um botão "Buscar por ID". Abaixo, há campos para Nome, Endereço, Telefone, Data de Nascimento, R.G., CPF e Sexo (com opções Masculino e Feminino). À direita, há uma seção "Restrição Entrada do Torcedor" com uma caixa de seleção "Sim" e um campo "Ingressos vendidos". No canto inferior direito, há dois botões: "Validar e Sair" e "Voltar".

Figura 23 - Validação da entrada do torcedor.

Fonte: Autoria própria.

A validação do ingresso acontece no momento em que é pressionado o botão “Salvar e Sair”: primeiro é verificado se o torcedor possui restrição à validação; se não possuir, então é feita uma comparação entre a capacidade do estádio e a

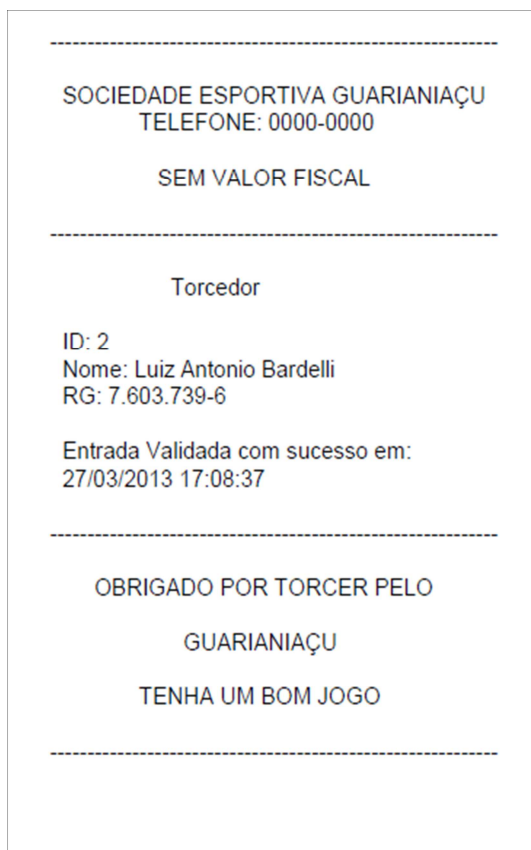
quantidade de ingressos vendidos; se o número de vendas for menor que a capacidade, a aplicação atualiza o campo "Acesso" do BD e valida a entrada do torcedor. Em seguida são chamadas as funções "MontaImpressão" e "Imprimir" pertencentes a classe "Val\_Ent".

A função que monta a impressão atribui em uma variável do tipo *string*, a formatação da impressão (figura 24). Esta função necessita ter um arquivo de texto (WriteLines.txt) de suporte para a impressão, onde são gravados todos os caracteres, para em seguida a função "Imprimir" fazer a transmissão dos dados para a impressora. O resultado da impressão é mostrado na figura 25.

```
225
226 private void montaImpressao()
227 {
228     string[] lines={
229         "-----",
230         "",
231         "SOCIEDADE ESPORTIVA GUARIANIAÇU",
232         "TELEFONE: 0000-0000",
233         "",
234         "SEM VALOR FISCAL",
235         "-----",
236         "",
237         "Torcedor",
238         "",
239         "ID: "+ ID,
240         "Nome: "+ l_Nome.Text,
241         "RG: "+l_RG.Text,
242         "",
243         "Entrada Validada com sucesso em: ",
244         "+ DateTime.Now,
245         "",
246         "-----",
247         "",
248         "OBRIGADO POR TORCER PELO",
249         "",
250         "GUARIANIAÇU",
251         "",
252         "TENHA UM BOM JOGO",
253         "",
254         "-----"};
```

Figura 24 - Montagem da impressão.

Fonte: Autoria própria.



**Figura 25 - Impressão resultante.**

**Fonte: Autoria própria.**

### 3.2.2.5 Limpar Acessos Para um Novo Jogo

Os torcedores que validaram a entrada para o último jogo, tiveram o campo “Acesso” atualizado para o valor um, e ao iniciar a validação da entrada para um novo jogo é necessário atualizar o campo “Acesso” de todos esses torcedores para o valor zero.

Quando é pressionado o botão “Limpar acessos para um novo jogo”, a classe “Autenticação” instância um novo objeto e abre uma janela (figura 26) para a autenticação por meio de uma senha de quatro algarismos, com valor padrão “1234”. Quando é inserida a senha corretamente esta classe faz a atualização com o valor zero no campo “Acesso” de todos os torcedores cadastrados no BD.

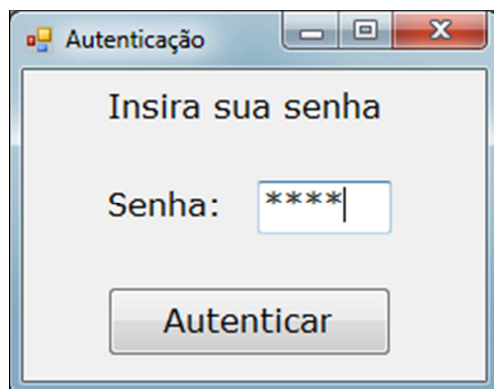


Figura 26 - Janela de Autenticação.

Fonte: Autoria própria.

### 3.2.3 Servidor das Catracas

Tal como a aplicação de cadastro e validação dos torcedores, o servidor das catracas é programado inteiramente na plataforma .NET, em linguagem C#. É composto de duas partes: a interface com o administrador do sistema e a interface com as catracas.

A função do servidor das catracas é fazer a intermediação entre o sistema de banco de dados e as catracas, ou seja, esse aplicativo obtém as informações armazenadas no banco de dados e verifica se o usuário possui acesso liberado ou não.

#### 3.2.3.1 Interface com as Catracas

O aplicativo de interface com as catracas é o cérebro de todo o sistema. Nele, são processados os dados das imagens enviadas pela catraca para que decisão de liberação seja realizada. A figura 27 mostra o fluxograma de funcionamento desta aplicação.

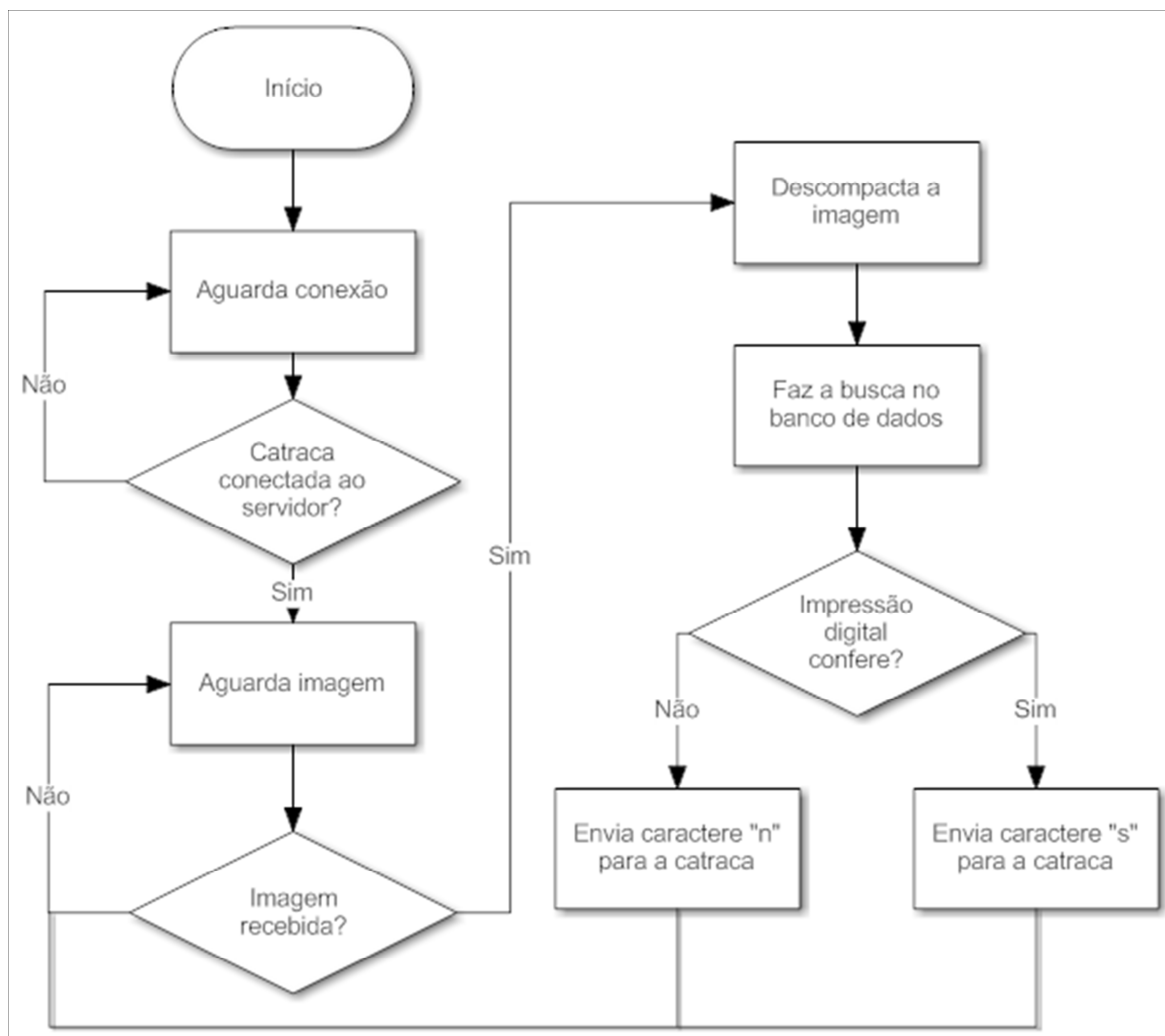


Figura 27 - Funcionamento básico do aplicativo de interface.

Fonte: Autoria própria.

Após ser ligado, o servidor aguarda conexões provenientes das catracas. Cada catraca se comunica através de um *socket* específico, e, como o IP do servidor é sempre o mesmo, cada catraca se conecta a uma porta específica. A porta 23 foi convencionada como padrão para a primeira catraca, e para cada catraca adicional, o número da porta é dado pelos números subsequentes a 23.

O servidor aguarda, então, o recebimento da imagem que será descompactada, como já citado anteriormente (item 3.2.2.1 – página 36). Assim, o servidor terá então uma imagem em formato *bitmap*, que será comparada com as impressões presentes no banco de dados.

Se for encontrada uma impressão digital que apresente similaridade suficiente com alguma impressão armazenada no BD, o servidor envia à controladora um

caractere “s” e apaga os dados desse usuário, para impedir que o torcedor utilize a entrada novamente. Se não houver uma impressão digital compatível, é enviado um caractere “n”. Quando esse processo termina, o servidor retorna ao estado de espera até o recebimento de uma nova imagem válida.

O *software* desenvolvido para essa interface apresenta quatro classes: *Server*, *Tcpserver*, *Fingerenv* e *Imag*. Um diagrama desta aplicação é mostrado na figura 28.

A partir da classe *Server* os objetos e métodos de outras classes são instanciados. Para o funcionamento do servidor é necessário o estabelecimento de conexão com as catracas e o recebimento de uma imagem. Isso é realizado a partir dos métodos *waitconnection()* e *receive()* da classe *Tcpserver*. Além de conter o método *response()* que envia a resposta a catraca, essa classe também define o *IP* utilizado pelo servidor. A porta utilizada é recebida como parâmetro quando o aplicativo é inicializado, e repassada a essa classe.

Após o recebimento da imagem, é realizada a descompactação e conversão para o formato *Bitmap*, através do método *CopyDataToBitmap()* pertencente a classe *Imag*.

A classe *Fingerenv* possui um método e quatro atributos. O método *match()* recebe uma imagem em formato *Bitmap*, então consulta um banco de dados próprio e decide se esta impressão digital confere com alguma das impressões que representam usuários com acesso liberado.

Todo esse mecanismo de busca e comparação de impressões digitais é desenvolvido com base no *Software Development Kit* (SDK) “IDKIT PC” do fabricante *Innovatrics*. Uma das funções desse SDK é criar um banco de dados próprio, onde é possível cadastrar impressões digitais para posterior consulta.

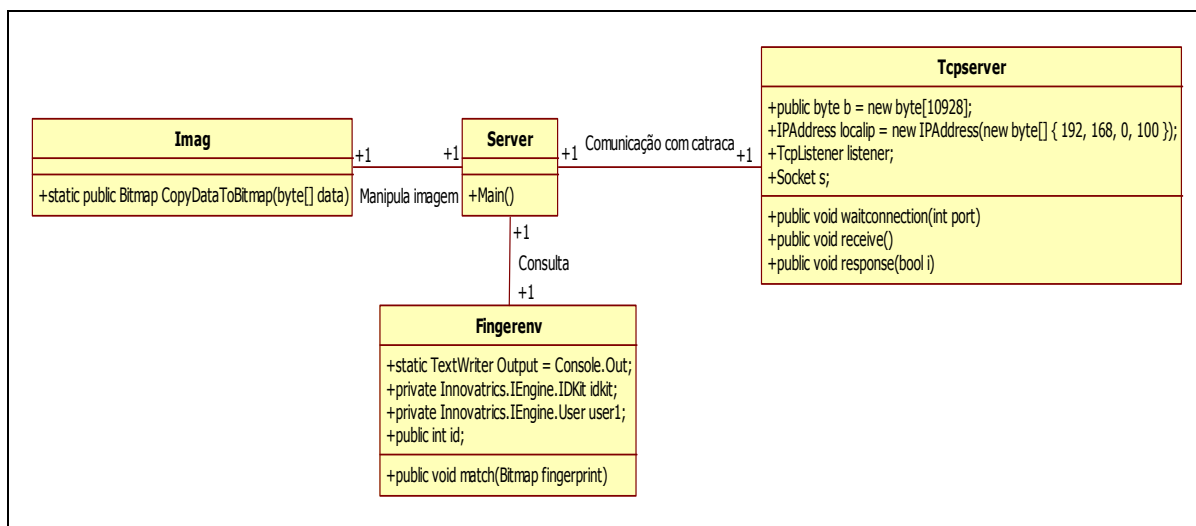


Figura 28 - Diagrama de classes da aplicação de interface com as catracas.

Fonte: Autoria própria.

A versão utilizada neste projeto é a “IDKIT PC FREE”. Nesta versão é possível criar um banco de dados de até 50000 usuários e utilizar um sistema de busca de um para n imagens. Um ponto relevante é o tempo necessário para liberação de acesso dos usuários. O SDK em questão garante tempos muito baixos de pesquisa e comparação em bancos de dados (tipicamente 50 milissegundos). A tabela 1 mostra a quantidade de impressões comparadas no intervalo de um segundo, em um banco de dados de um milhão de usuários e utilizando um processador *Intel Core2 Duo*. Essa tabela é fornecida pelo fabricante.

Tabela 1 - Velocidade de comparação de impressões digitais.

Fonte: SUPREMA, 2012.

Speed level	Max.Rotation=30	Max.Rotation=50	Max.Rotation=180
1	300	300	300
2	30500	26000	12000
3	137000	105000	40000
4	222000	184000	89000
5	340000	298000	181000
6	490000	403000	185000
7	632000	547000	275000
8	747000	660000	360000

A quantidade de impressões comparadas leva em conta também atributos como a rotação da imagem e o nível de velocidade. O IDKIT PC FREE permite níveis de velocidade entre um e oito e rotação de imagem de até 180 graus. Como o banco de dados é de no máximo 50000 pessoas, foi utilizado o valor máximo de rotação de 30 graus com nível de velocidade três. Assim é garantido, mesmo no pior caso, 137000 comparações por segundo, ou seja, o tempo máximo de busca e comparação dentro do banco de dados cheio é de aproximadamente 360 milissegundos, que representa uma parte ínfima do tempo total de acesso.

A confiabilidade do sistema é determinada por um atributo chamado *Threshold*, presente no SDK. A tabela 2 mostra as falsas taxas de aceitação para os diferentes valores de *Threshold*.

O projeto utiliza o valor de 12300 de *Threshold*, portanto a falsa taxa de aceitação, ou FAR, tem valor menor que 0.001%.

**Tabela 2 - Confiabilidade do sistema.**

Fonte: SUPREMA, 2012.

FAR	Threshold
1%	9600
0,1%	10200
0,01%	11000
0,001%	12100

Durante a inicialização do servidor, na interface com o usuário, as impressões digitais e os números de identificação dos torcedores com acesso liberado são copiados para a base de dados a ser utilizada pelo "IDKIT PC". O método *match()* conecta-se através do "IDKIT PC" à essa base de dados e, a partir da rotina *Find()*, também pertencente ao "IDKIT PC", busca uma impressão digital compatível.

O código de retorno desse método é o número de identificação do usuário. Portanto, se retornar o número zero, não foi encontrada nenhuma impressão digital compatível.



### 3.2.3.1 Interface com o Administrador do Sistema

O aplicativo “Administrador do Sistema” (figura 29) tem como objetivo fazer a interface entre o operador do sistema e controlar os diversos aplicativos que fazem a interface com as catracas. É possível também fazer a cópia dos dados do banco de dados central para o banco utilizado pelo “IDKIT PC”.

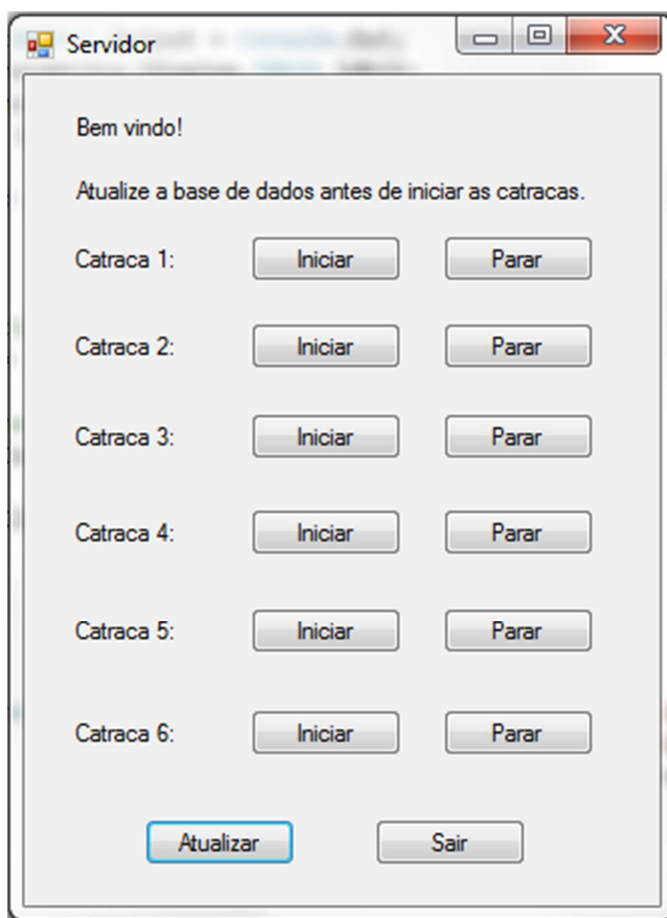


Figura 29 - Aplicação de interface com o usuário.

Fonte: Autoria própria.

A partir dessa interface, é possível abrir um aplicativo específico para cada catraca. Isso foi desenvolvido levando em conta uma melhor divisão de tarefas em relação ao uso do processador, já que cada aplicativo será tratado como um processo a parte, o que melhora o desempenho do programa.

Após a abertura do programa é realizada a atualização da base de dados. Quando ocorre o clique no botão “Atualizar”, a aplicação conecta-se ao banco de

dados principal e faz a cópia apenas das informações dos torcedores com acesso liberado. Esse procedimento é realizado através do método *sync()*.

Para a cópia dos dados do servidor de banco de dados central para o servidor das catracas, primeiro é necessária a conexão com o BD. Isso é feito através de uma biblioteca específica da plataforma .NET. Após a conexão, é enviado um comando que representa um *stored procedure*, através do qual o banco de dados executa uma tarefa buscando apenas os usuários que possuem o campo acesso preenchido. Com esses usuários selecionados, é executado o método *RegisterUserAs()*, do “IDKIT PC”, com os parâmetros ID e imagem.

Atualizada a base de dados, o administrador do sistema precisa iniciar o serviço correspondente a cada catraca clicando no botão “Iniciar”. É então iniciada uma nova aplicação de interface com as catracas, que obedece ao parâmetro que indica o número da porta.

#### 3.2.4 Servidor de Banco de Dados

O sistema de banco de dados do projeto foi criado para armazenar as informações dos torcedores. Além das informações básicas do torcedor, estão contidas as impressões digitais, fotos, e identificadores de acesso. O gerenciador foi programado com base na linguagem SQL (figura 30).

Dentro dos requisitos do projeto, foram consideradas como as principais informações dos torcedores:

- Número de identificação, ou ID;
- Nome;
- Endereço;
- Telefone;
- R.G;
- CPF;
- Data nascimento;
- Sexo;
- Foto;
- Impressão Digital.
- Restrição

Para cada torcedor existe, ainda, uma variável chamada “Acesso”, que define se o torcedor teve a entrada ao estádio validada ou não. Através da consulta dessa variável é realizada a cópia dos dados apenas dos torcedores que tenham essa variável com valor um, ou seja, acesso liberado, essa cópia é do servidor de banco de dados para a aplicação servidora das catracas. Outro requisito importante é que, dentre todos esses campos pertencentes à tabela, apenas o campo destinado à restrição de acesso pode estar nulo, o que foi garantido durante a criação da tabela. Também há a garantia de que o torcedor somente será incluso na base de dados se todos os campos forem preenchidos.


	Column Name	Data Type	Allow Nulls
	ID	int	<input type="checkbox"/>
	Nome	varchar(50)	<input type="checkbox"/>
	Endereco	varchar(50)	<input type="checkbox"/>
	Telefone	varchar(50)	<input type="checkbox"/>
	RG	varchar(50)	<input type="checkbox"/>
	CPF	varchar(50)	<input type="checkbox"/>
	Nasc	varchar(50)	<input type="checkbox"/>
	Sexo	int	<input type="checkbox"/>
	Foto	image	<input type="checkbox"/>
	Digital	image	<input type="checkbox"/>
	Acesso	int	<input type="checkbox"/>
	Restri	int	<input type="checkbox"/>
	Tipores	varchar(150)	<input checked="" type="checkbox"/>

Figura 29 - Tabela SQL de informações de torcedores.

Fonte: Autoria própria.

É criada também, para funcionamento concomitante com o servidor das catracas, uma chamada *stored procedure* de nome “ids liberados”, no qual, através de um código SQL, é possível selecionar apenas os torcedores que tenham acesso liberado.

## 4 RESULTADOS OBTIDOS

Este capítulo mostra os resultados obtidos ao longo do projeto. Apesar de o projeto ser apenas um protótipo, o sistema demonstrou a robustez e confiabilidade necessária para aplicação em situações reais.

### 4.1 ADMINISTRAÇÃO DO SISTEMA

As aplicações de cadastro e interface com as catracas ficaram em um formato intuitivo e de fácil utilização, o que minimiza o tempo de treinamento e facilita a administração de todo o sistema.

Todos os botões da aplicação de cadastro indicam a ação a ser tomada. Um dos grandes benefícios desta aplicação é a possibilidade de impedir que determinado torcedor compre o ingresso, através da opção de restrição. Este artifício visa diminuir a violência dentro de estádios.

A aplicação que faz a interface apresenta um leiaute claro e simples, no intuito de diminuir a possibilidade de erros de operação.

Uma vez configurado, o servidor de banco de dados inicia sua execução no momento em que a máquina liga, ou seja, administrador do sistema não precisa em nenhum momento ter alguma preocupação com este.

### 4.2 TEMPO DE LIBERAÇÃO

O ponto mais crítico neste projeto é o tempo decorrido desde a captura da impressão digital do usuário até a liberação de acesso. Uma grande espera, em uma situação real, gera tumultos.

Durante o desenvolvimento do projeto ao aplicar tecnologias eficazes, se obteve um tempo muito bom de três segundos em média. Porém, ainda assim, há espaço para modificações e aprimoramentos no projeto.

O gargalo de tempo ocorre na própria catraca. O tempo para a imagem ser transmitida do sensor para a placa controladora é um pouco superior a um segundo. Até ser processada e enviada, a imagem leva pouco mais de dois segundos até

atingir o servidor. Por sua vez, o servidor é rápido na validação de acesso do usuário e apresenta sua resposta em menos de segundo.

Uma característica interessante do projeto é que, independente do tamanho do banco de dados, o tempo de acesso não tende a mudar. Isto é devido à escolha dos parâmetros de configuração (máxima rotação da imagem e velocidade de identificação, vide tabela 2), que foi realizada levando em conta um melhor desempenho. Além disso, como é aberto um processo servidor para cada catraca, é feito um melhor uso do processador. Segundo o fabricante, no pior caso, a identificação do usuário no servidor ocorre em 360 milissegundos.

### 4.3 CONFIABILIDADE

A confiabilidade é medida na falsa taxa de aceitação. Com os valores utilizados no programa, essa taxa fica em torno de 0,001%. Ou seja, há 0,001% de probabilidade de um torcedor não cadastrado conseguir acesso ao estádio. Ainda que esse valor seja bastante baixo, em um banco de dados muito grande, ele pode se tornar significativo.

Há diversas formas de melhorar ainda mais a confiabilidade. Buscando diminuir os bancos de dados de cada catraca, é possível a criação de setores específicos. Ainda, há a possibilidade de alteração nos atributos do programa, para que este fique mais confiável, porém, isso pode aumentar a taxa de falsa rejeição, causando transtornos aos torcedores.

### 4.3 CUSTO

A plataforma é de baixo custo, quando comparada a outros equipamentos similares. A tabela 3 mostra todos os custos para um projeto com uma catraca.

Tabela 3 - Custos da Catraca.

Fonte: Autoria própria.

Item	Quantidade	Valor Unit.	Valor Total
Sensor Biométrico	2	R\$ 250,00	R\$ 500,00
Mecanismo catraca	1	R\$ 1.000,00	R\$ 1.000,00
Kit LPCXpresso	1	R\$ 400,00	R\$ 400,00
Componentes	1	R\$ 20,00	R\$ 20,00
Placa circuito impresso	1	R\$ 20,00	R\$ 20,00
Computadores	2	R\$ 2.000,00	R\$ 4.000,00
Adaptador serial-USB	1	R\$ 30,00	R\$ 30,00
Roteador	1	R\$ 50,00	R\$ 50,00
Licença software	1	R\$ 800,00	R\$ 800,00
Valor Total			R\$ 6.820,00

O maior custo é com a compra de computadores. Se o cliente já possuir computadores que sejam capazes de executar a aplicação, o custo se reduz a R\$ 2.820,00.

Mesmo que não exista uma solução idêntica no mercado, esse custo é baixo se comparado a produtos similares de outros fabricantes. O fabricante AKLEMAQ fornece um sistema de catraca biométrica por R\$ 4.500,00. 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. PLANO DE NEGÓCIOS

### 5.1 DEFINIÇÃO DO PRODUTO

#### 5.1.1 Natureza do Negócio

A empresa visa à 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 suma importância. A busca pelo uso de novas tecnologias, que tornem os produtos mais eficazes e eficientes, é constante. O projeto e a montagem, incluindo *software* e *hardware*, 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 garantem segurança ao negócio do cliente.

#### 5.1.2 Missão

Ser um fabricante e prestador de serviços de respeito na América Latina. Buscar sempre a inovação, nas mais diferentes maneiras, e assim trazer soluções que resolvam dificuldades no negócio do cliente.

#### 5.1.3 Valores

Almejar o sucesso é o principal propósito da empresa, porém, acima de tudo, com integridade e ética. E esse sucesso virá com a dedicação ao sucesso de cada cliente.

#### 5.1.4 Visão

Ser referência no mercado de sistemas de controle de acesso na América Latina, e tomar posição entre as três maiores empresas do setor até 2017.

#### 5.1.5 Objetivos

- Desenvolver novas soluções até 2015;
- Expandir o mercado para a América Latina até 2014;
- Criar uma primeira filial internacional até 2016;
- Estabelecer parcerias em outros continentes até 2025.

### 5.2 DEFINIÇÃO DO PRODUTO

#### 5.2.1 Descrição

O PSCBEF é uma solução eletrônica de controle de acesso a estádios de futebol utilizando biometria. Através de aplicações específicas, é possível fazer o cadastro e manipular as catracas de maneira a garantir o acesso dos torcedores às dependências dos estádios de futebol.

A solução conta diversas funcionalidades que visam o aumento da segurança em estádios de futebol. Além do cadastro das informações pessoais básicas dos torcedores, há o cadastro de foto, impressão digital e, em caso de necessidade, de restrições quanto ao acesso. O uso da biometria para o acesso promove um controle mais eficaz, pois, devido às propriedades das impressões digitais, não existe a possibilidade de transferência de ingresso sem o conhecimento do administrador do estádio.

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.

### 5.2.2 Tecnologia e Descrição Técnica

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 minibloqueio fabricado pela Wolpac. Ainda na catraca, há uma placa controladora com um processador ARM Cortex M3, de alto desempenho.

Ainda, há duas aplicações, uma para interface com as catracas e outra para cadastro e validação de usuários. Ambas funcionam em sistema operacional Windows e foram desenvolvidas na plataforma .NET.

O sistema gerenciador de banco de dados é o Microsoft SQL Server 2012. Após a configuração, esse banco de dados é responsável pelo armazenamento de todas as informações dos torcedores.

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



### 5.2.3 Vantagens Competitivas

O PSCBEF não possui concorrentes diretos, mas sim produtos e soluções alternativas, pois nenhum concorrente é especialmente designado para estádios de futebol. Além disso, a maioria dos sistemas de controle de acesso não utiliza biometria, e os que utilizam, possuem limitações quanto ao cadastro, custo, tamanho do banco de dados e facilidade de operação.

Em geral, catracas que possuem o sistema de biometria, são muito limitadas quanto ao cadastro, já que todo o banco de dados contendo as impressões digitais é armazenado na própria catraca. O PSCBEF 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 uso da tecnologia menos oneroso.

## 5.3 DEFINIÇÃO DO MERCADO

### 5.3.1 Segmentação do Mercado

Todo estádio de futebol precisa de um controle de acesso de torcedores. E é nesse nicho de mercado que o PSCBEF se encontra.

### 5.3.2 Segmento Alvo

Inicialmente, serão visados clientes de menor porte, ou seja, clubes de menor porte que desejam modernizar o seu sistema de controle de entrada as dependências do estádio.

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

### 5.3.3 Tamanho

Todos os clubes de futebol do Brasil e exterior são clientes em potencial. Apesar desse grande mercado, não é necessariamente o grande volume de vendas que trará grande retorno financeiro, e sim a venda aliada a prestação de serviços técnicos associados ao sistema.

### 5.3.4 Necessidade

A segurança e conforto dos torcedores são essenciais nas dependências dos estádios de futebol. E são exatamente esses dois pontos que o PSCBEF procura melhorar. Sem a necessidade de um ingresso em papel, o acesso do torcedor fica mais cômodo, já que é necessária apenas a impressão digital para a entrada ao estádio. Além disso, como todos os torcedores serão cadastrados e aqueles com restrição de entrada não terão acesso, há uma maior segurança e controle dos presentes nos estádios.

### 5.3.5 Crescimento

Espera-se que a necessidade para serviços de controle de acesso aumente, pois, além da necessidade constante de aumento da segurança e conforto, o número de torcedores aumenta a cada dia.

### 5.3.6 Cliente Alvo

O produto tem potencial para se tornar líder em seu mercado, este, que apesar de restrito, é muito promissor. De início, espera-se ter como clientes pequenos clubes, porém com o passar do tempo, pretende-se atingir grandes clubes.

### 5.3.7 Aplicação

Controle de acesso a estádios de futebol utilizando biometria.

## 5.4 ESTRATÉGIA DE MARKETING

### 5.4.1 Preço

O cliente comprará o produto e ficará isento de taxa de instalação. Se acaso desejar, o cliente pode adquirir serviços personalizados de manutenção e suporte, mesmo dentro do prazo de garantia.

### 5.4.2 Publicidade

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

### 5.4.3 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.

### 5.4.4 Instalação

A empresa contará com uma sede bem localizada, em um bairro próximo ao centro da cidade de Curitiba. Tal sede precisará de três salas de projeto, cada uma com três computadores, uma bancada de testes e outra para montagem das placas. Necessitará também uma oficina para serviços mais pesados, uma sala de reunião e uma sala para a gerência.

## 5.5 ESTRUTURA ORGANIZACIONAL

### 5.5.1 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.

## 5.5.2 Infraestrutura

### 5.5.2.1 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, na oficina, já que a demanda por catracas não é muito grande.

As placas são montadas por parceiros, empresas de confecção de placas de circuito impresso. Dentro dessa atribuição, existem duas empresas de médio porte em Curitiba, e uma de grande porte em Santa Catarina, que podem facilmente suprir as necessidades.

### 5.5.2.2 Fornecedores

As placas de circuito impresso serão projetadas na sede da empresa, e serão produzidas em Curitiba pela CIRCUITEL e CIRCUIBRAS. O sensor biométrico e o processador serão importados, da China e Suécia, respectivamente. Os componentes mais comuns, como cabos e conectores, serão comprados em empresas nativas da cidade, como a BETA COMERCIAL e a PARESLACK.

### 5.5.2.3 Logística

Todos os produtos que serão adquiridos para a montagem serão entregues na sede da empresa. A montagem será feita e os produtos serão levados aos

clientes através de empresas transportadoras. A equipe técnica trabalhará na instalação do produto para o cliente.

## 5.6 PLANO FINANCEIRO

### 5.6.1 Contabilização

A tabela 4 mostra a contabilização de mobiliário e equipamento da sede da empresa e a tabela 5 os gastos iniciais.

**Tabela 4 - Mobiliário e Equipamento da Sede da Empresa.**

Fonte: Autoria própria.

<b>Item</b>	<b>Quantidade</b>	<b>Valor Unit.</b>	<b>Valor Total</b>
Computador	10	R\$ 2.000,00	R\$ 20.000,00
Bancada	2	R\$ 1.000,00	R\$ 2.000,00
Osciloscópio	2	R\$ 2.500,00	R\$ 5.000,00
Estação de solda	5	R\$ 150,00	R\$ 750,00
Soprador de ar quente	1	R\$ 450,00	R\$ 450,00
Kit de ferramentas	5	R\$ 400,00	R\$ 2.000,00
Impressora	1	R\$ 300,00	R\$ 300,00
Serra elétrica	1	R\$ 250,00	R\$ 250,00
Perfuradora	1	R\$ 400,00	R\$ 400,00
Material de oficina	1	R\$ 800,00	R\$ 800,00
Material de escritório	1	R\$ 500,00	R\$ 500,00
Bancada para oficina	1	R\$ 1.000,00	R\$ 1.000,00
Iluminação articulada	3	R\$ 200,00	R\$ 600,00
Mesas de escritório	4	R\$ 300,00	R\$ 1.200,00
Cadeira de escritório	15	R\$ 200,00	R\$ 3.000,00

Tabela 5 - Gastos Iniciais.

Fonte: Autoria própria.

Aluguel de imóvel	R\$ 3.200,00
Limpeza	R\$ 450,00
Instalação dos móveis e equipamentos	R\$ 400,00
Pintura	R\$ 850,00
Reparos	R\$ 1.000,00
Seleção e contrato de funcionários	R\$ 1.800,00
Taxas de abertura de Microempresa	R\$ 3.000,00
Material para primeira produção	R\$ 7.000,00
Valor Total	R\$ 17.700,00

## 5.6.2 Custo

Para o funcionamento da empresa, é necessário o levantamento dos custos mensais, mostrado na tabela 6.

Tabela 6 - Gastos Mensais.

Fonte: Autoria própria.

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 Estagiários	2	R\$ 800,00	R\$ 1.600,00
Salário Técnicos	1	R\$ 1.550,00	R\$ 1.550,00
Salário Faxineira	1	R\$ 800,00	R\$ 800,00
Salário Secretária	1	R\$ 900,00	R\$ 900,00
Aluguel	1	R\$ 3.200,00	R\$ 3.200,00
Conta de Luz	1	R\$ 410,00	R\$ 410,00
Conta telefone/internet	1	R\$ 200,00	R\$ 200,00
Conta de Água	1	R\$ 200,00	R\$ 200,00
		Valor Total	R\$ 24.260,00

### 5.6.3 Fluxo de Caixa Esperado

Através de uma análise da expectativa de fluxo de caixa (Tabela 7) para os primeiros 12 meses é possível constatar que com um investimento inicial de R\$200 mil e firmando parcerias com 10 clientes, 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 projeto e 1000 reais pelos serviços mensais de manutenção. O custo mensal de manutenção da empresa é calculado em R\$24.260,00.

Tabela 7 - Fluxo de caixa.

Fonte: Autoria própria.

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\$ 24.260,00 R\$ 800,00 R\$ 300,00	R\$ 45.440,00
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\$ 24.260,00 R\$ 300,00	R\$ 27.800,00
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\$ 24.260,00 R\$ 300,00	R\$ 43.360,00
9	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\$ 24.260,00 R\$ 300,00	R\$ 70.920,00
8	1 - Mensalidade do contrato de manutenção 2 - Mensalidade do contrato de manutenção 3 - Terceira venda	R\$ 1.000,00 R\$ 1.000,00 R\$ 40.000,00	1 - Gastos mensais 2 - Instalação do sistema 3 - Custo de manutenção	R\$ 24.260,00 R\$ 800,00 R\$ 300,00	R\$ 32.480,00
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\$ 24.260,00 R\$ 300,00	R\$ 75.840,00
6	1 - Primeiro contrato de Manutenção	R\$ 1.000,00	1 - Gastos mensais 2 - Custo de manutenção	R\$ 24.260,00 R\$ 300,00	R\$ 59.400,00
5	1 - Primeira venda	R\$ 30.000,00	1 - Gastos mensais 2 - Instalação do sistema	R\$ 24.260,00 R\$ 800,00	R\$ 82.960,00
4		R\$ 0,00	1 - Gastos mensais	R\$ 24.260,00	R\$ 78.020,00
3		R\$ 0,00	1 - Gastos mensais 2 - Produção das placas	R\$ 24.260,00 R\$ 1.500,00	R\$ 102.280,00
2		R\$ 0,00	1 - Gastos mensais 2 - Compra de materiais	R\$ 24.260,00 R\$ 30.000,00	R\$ 128.040,00
1	1 - Investimento Inicial	R\$ 200.000,00	1 - Custos iniciais	R\$ 17.700,00	R\$ 182.300,00
Mês					
receitas					
Despesas					
Acumulado					



## 6 CONCLUSÃO

Neste trabalho de conclusão de curso foi apresentado um sistema de controle de acesso a estádios de futebol utilizando biometria. Buscando a velocidade e confiabilidade do sistema, foi demonstrado que é possível o uso de impressões digitais como forma de acesso aos estádios.

Apesar de já existir, no mercado brasileiro, soluções de controle de acesso via biometria, nenhuma é concebida e completamente focada no uso em estádios de futebol.

O ponto central da ideia do projeto é o aumento da segurança e do conforto dos torcedores. O acesso via impressão digital resolve os principais inconvenientes presentes no sistema de acesso via *smartcard*. O primeiro é a impossibilidade de transferência de ingresso, prática comum entre os torcedores. O segundo problema que o PSCBEF resolve é a revenda de ingressos e a falsificação, que normalmente ocorre em jogos de maior apelo, através de cambistas que operam com preços abusivos.

O principal benefício deste projeto é que o clube tem o conhecimento de todos os torcedores presentes do estádio e, em caso de alguma ocorrência, tem uma base de dados completa para posteriormente identificar a pessoa envolvida. A comodidade é outro ponto a ser ressaltado: o torcedor não precisa se preocupar com o ingresso físico, já que a credencial de acesso estará sempre à disposição; mesmo assim é impresso um comprovante no momento da validação da entrada.

### 6.1 CONSIDERAÇÕES SOBRE O DESENVOLVIMENTO DO PROJETO

O desenvolvimento de todo o sistema foi certamente um grande desafio aos integrantes da equipe. Em determinados momentos fez-se necessário o aprendizado de tecnologias completamente novas aos membros da equipe, o que tornou o processo de elaboração do projeto mais oneroso.

Era sabido, desde a concepção inicial do projeto, que em geral sistemas que empregam biometria não são triviais. Porém, empregando ferramentas adequadas como um SDK, um sensor específico e um processador rápido, a manipulação das impressões digitais seria mais simples, resultando em um sistema confiável, estável e veloz.

O *software* e *firmware* foram desenvolvidos em um processo iterativo e incremental. Cada código começou com versões simples de subconjuntos de requerimentos, e a cada iteração novas funcionalidades foram adicionadas. Assim, houve a evolução gradual até tal ponto em que as aplicações contemplaram todos os requerimentos.

Ainda, o *software* foi totalmente desenvolvido em linguagem C#, que foi estudada com afinco, já que até então, era desconhecida para os membros da equipe. Além disso, foi necessário também o estudo sobre bancos de dados e linguagem SQL, já que ambos são conceitos base desse projeto. Ainda que tenham tornado o projeto mais custoso, foi muito proveitoso o aprendizado dessas outras tecnologias.

O *firmware* do controlador das catracas foi escrito em linguagem de programação C e a escolha do processador *Cortex M3*, juntamente com a placa base de desenvolvimento, se mostrou vantajosa pois este conjunto possui os periféricos necessários para o desenvolvimento do projeto, além da grande velocidade de processamento.

## 6.2 OBJETIVOS ALCANÇADOS

O principal objetivo, de comprovar a viabilidade do uso de impressões digitais no controle de acesso a estádios de futebol, foi atingido. Através do uso de um SDK específico e um sistema gerenciador de banco de dados, foi criado um sistema rápido e confiável.

A confiabilidade é dada por parâmetros do SDK. A partir de parâmetros específicos, escolhidos dentro das necessidades do projeto, foi garantida uma taxa de falsa aceitação é de 0.001%, ou seja, o sistema é bastante confiável.

Outro ponto foi o tempo necessário para a liberação do usuário. Neste quesito, o gargalo está na transmissão dos dados pela porta serial, entre o módulo biométrico e o processador, que é realizada com uma taxa de 115200bps. Somado ao tempo necessário para a comunicação via *Ethernet* entre a catraca e o servidor, o tempo para a transmissão da imagem até o servidor ficou em aproximadamente 2 segundos.

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

Todas essas operações, a partir da captura da digital na catraca, transmissão para o servidor, comparação da imagem e retorno da informação, liberando ou não o torcedor, ficaram em aproximadamente três segundos.

As aplicações, tanto de cadastro como de interface com as catracas, são intuitivas e confiáveis. Ambas tem um bom desempenho, mesmo em máquinas com menor capacidade computacional.

Todas as funcionalidades propostas na especificação de requisitos, foram implementadas e apresentaram bom desempenho. Desta forma, apesar do rótulo de protótipo, devido às características demonstradas, é possível dizer que o projeto em questão já é uma solução viável para o mercado.

## REFERÊNCIAS

ALFORD, Ryan. C# Corner. **Communicating with Serial Port in C#**. Dezembro de 2008. Disponível em: < <http://www.c-sharpcorner.com/uploadfile/eclipsed4utoo/communicating-with-serial-port-in-C-Sharp> >. Acesso em 01/03/2013.

ALVES, Marcus. ESPN: atualizado em: 09/03/2013. Disponível em: < <http://esportes.br.msn.com/futebol/libertadores/noticias/story.aspx?cp-documentid=256668596>>. Acesso em: 29/03/2013.

BAZEN, Asker M., et al. **A Correlation-Based Fingerprint Verification System**. Universidade do Twent, Holanda. Disponível em: < <http://eprints.eemcs.utwente.nl/1776/01/009-2000-Bazen-Verwaaijen-etal.pdf> >. Acesso em 03/04/2013.

CANEDO, José Alberto. Fórum Biometria. Disponível em: <<http://www.forumbiometria.com/biometria.html>>. Acesso em 02/04/2013.

CBF. Confederação Brasileira de Futebol. Disponível em: < [www.cbf.com.br](http://www.cbf.com.br)>. Acesso em: 29/03/2013.

EMBEDDED Artists. **LPCXpresso Base Board User's Guide, ver B**. Disponível em: <[http://www.embeddedartists.com/sites/default/files/support/xpr/base/LPCXpresso\\_BaseBoard\\_rev\\_B\\_Users\\_Guide.pdf](http://www.embeddedartists.com/sites/default/files/support/xpr/base/LPCXpresso_BaseBoard_rev_B_Users_Guide.pdf) >. Acesso em 11/12/2012.

EMBEDDED Artists. **LPCXpresso Base Board Schematic**. Disponível em: < [http://www.cs.umd.edu/class/fall2012/cmsc498a/manuals/lpcxpresso\\_lpc1769\\_schematic.pdf](http://www.cs.umd.edu/class/fall2012/cmsc498a/manuals/lpcxpresso_lpc1769_schematic.pdf) >. Acesso em 11/12/2012.

FBI. Federal Bureau of Investigation. **Fingerprint Identification**. Disponível em: < [http://www.fbi.gov/about-us/cjis/fingerprints\\_biometrics/fingerprint-overview](http://www.fbi.gov/about-us/cjis/fingerprints_biometrics/fingerprint-overview) >. Acesso em: 03/04/2013.

HARRIS, Tom. **Como funcionam os leitores de impressões digitais**. Disponível em: < <http://informatica.hsw.uol.com.br/leitores-de-impressoes-digitais.htm> >. Acesso em: 02/04/2013.

HENRY, E. R. **Classification and Uses of Finger Prints**. Disponível em: < <http://www.clpex.com/Information/Pioneers/henry-classification.pdf> >. Acesso em: 02/04/2013.

KIOSKEA .net. **Os Modelos de SGBD**. Disponível em: < <http://pt.kioskea.net/contents/bdd/bddtypes.php3> >. Acesso em 26/02/2013.

MACORATTI, José Carlos. **Cadastro de Alunos com Foto no SQL Server**. Disponível em: < <http://www.macoratti.net> >. Acesso em: 22/02/2013.

MACORATTI, José Carlos. **Salvando e Recuperando Imagens do SQL Server**. Disponível em: < <http://www.macoratti.net> >. Acesso em: 20/02/2013.

MAZUMDAR, Subhra; DHULIPALA, Venkata. **Biometric Security Using Finger Print Recognition**. Universidade da California, San Diego. Disponível em: < [http://cseweb.ucsd.edu/classes/fa08/cse237a/finalproject/smazumdar\\_final\\_report.pdf](http://cseweb.ucsd.edu/classes/fa08/cse237a/finalproject/smazumdar_final_report.pdf) >. Acesso em: 02/04/2013.

MINISTERIO do esporte. Torcida Legal. Disponível em: < <http://www.esporte.gov.br/assessoriaEspecialFutebol/torcidaLegal/default.jsp> >. Acesso em 01/04/2013.

MSDN. Microsoft. Fórum. Disponível em: < <http://social.msdn.microsoft.com/Forums> >. Acesso em 20/02/2013.

MSDN. Microsoft. **Getting Started with the .NET Framework**. Disponível em: < <http://msdn.microsoft.com/en-us/library/hh425099.aspx> >. Acesso em 27/02/2013.

MSDN. Microsoft. **Visual C#**. Disponível em: < <http://msdn.microsoft.com/pt-br/library/vstudio/kx37x362%28v=vs.100%29.aspx> >. Acesso em 27/02/2013.

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

NIST. National Institute of Standards and Technology. Disponível em: < [http://www.nist.gov/public\\_affairs/licweb/fingerprints.htm](http://www.nist.gov/public_affairs/licweb/fingerprints.htm) >. Acesso em 02/04/2013.

NXP Semiconductors. **LPC 1769 Data Sheet. rev. 9.** Disponível em: < [http://www.nxp.com/documents/data\\_sheet/LPC1769\\_68\\_67\\_66\\_65\\_64\\_63.pdf](http://www.nxp.com/documents/data_sheet/LPC1769_68_67_66_65_64_63.pdf) >. Acesso em: 11/12/2012.

NXP Semiconductors. **LPC 17XX User Manual rev. 2.** Disponível em: < [http://www.nxp.com/documents/user\\_manual/UM10360.pdf](http://www.nxp.com/documents/user_manual/UM10360.pdf) >. Acesso em: 11/12/2012.

ORACLE faq's. Disponível em: < [http://www.oraFAQ.com/faq/what\\_are\\_the\\_difference\\_between\\_ddl\\_dml\\_and\\_dcl\\_commands](http://www.oraFAQ.com/faq/what_are_the_difference_between_ddl_dml_and_dcl_commands) >. Acesso em 27/02/2013.

PAPILOSCOPIA. **Estudo da Pele.** Disponível em: < [http://www.papiloscopia.com.br/estudo\\_das\\_papilas.html](http://www.papiloscopia.com.br/estudo_das_papilas.html) >. Acesso em 02/04/2013.

PRABHAKAR, Salil; JAIN, Anil. **Fingerprint Identification.** Disponível em: < <http://biometrics.cse.msu.edu/fingerprint.html> >. Acesso em 03/04/2013

PULCINELLI, Márcio. **Acessando a WebCam com C#.** Disponível em: < <http://blog.marcio-pulcinelli.com/2011/06/05/acessando-webcam-com-c/> >. Acesso em 19/02/2013.

RANJAN. Code Project. **Serial Communication in C#.** Novembro de 2007. Disponível em: < <http://www.codeproject.com/Articles/19256/Serial-Communication-in-C> >. Acesso em 01/03/2013.

RIBEIRO, Diego. Rede Globo. Disponível em: < <http://globoesporte.globo.com/futebol/times/corinthians/noticia/2013/02/torcedor-boliviano-morre-atingido-por-sinalizador-diz-policia-local.html> >. Acesso em: 29/03/2013.

SUPREMA. **UniFinger SFM3520-OP Data Sheet, version 1.0.** Disponível em: < [http://www.supremainc.com/eng/Datasheet/UF\\_SFM3520OP\\_Datasheet\\_v1.0.pdf](http://www.supremainc.com/eng/Datasheet/UF_SFM3520OP_Datasheet_v1.0.pdf) >. Acesso em 11/12/2013.

SUPREMA. **Packet Protocol Manual, version 2.5.** Disponível em: < [http://vistas-gmbh.com/fileadmin/Downloads/Prospekte/Leseprobe%20UF\\_Protocol\\_Manual\\_V2.5.pdf](http://vistas-gmbh.com/fileadmin/Downloads/Prospekte/Leseprobe%20UF_Protocol_Manual_V2.5.pdf) >. Acesso em: 11/12/2012.

TEXAS Instrumensts. **Cortex – M3/M4F Instruction Set**. Disponível em: <  
<http://www.ti.com/lit/ug/spmu159a/spmu159a.pdf> >. Acesso em: 11/12/2012.