

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DO SERTÃO PERNAMBUCANO
CURSO DE GESTÃO DA TECNOLOGIA DA INFORMAÇÃO**

DANIELA NOGUEIRA FERRAZ CORNÉLIO

**Elaboração de um protótipo de ponto eletrônico
utilizando Arduino e RFID: Uma visão para o Instituto
Federal de Educação, Ciência e Tecnologia do Sertão
Pernambucano – Campus Floresta**

Floresta

2016

DANIELA NOGUEIRA FERRAZ CORNÉLIO

**Elaboração de um protótipo de ponto eletrônico
utilizando Arduino e RFID: Uma visão para o Instituto
Federal de Educação, Ciência e Tecnologia do Sertão
Pernambucano – Campus Floresta**

Monografia apresentada ao curso de Gestão de Tecnologia da Informação, do Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano - Campus Floresta, como requisito parcial para obtenção do grau de Tecnólogo em Gestão de Tecnologia da Informação.

Orientador: Ezio Alves Oliveira

Co-orientador: Paulo Thiago Lima do Nascimento

Floresta

2016

C814e Cornélio, Daniela Nogueira Ferraz.

Elaboração de um protótipo de ponto eletrônico utilizando Arduino e RFID: uma visão para o Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano – Campus Floresta. / Daniela Nogueira Ferraz Cornélio – 2016.

73f. il.

Trabalho de Conclusão de Curso (Tecnólogo em Gestão de Tecnologia da Informação) – Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano – Campus Floresta. Floresta, 2016.

Orientação: Eziom Alves Oliveira .

1. Ponto eletrônico . 2. Arduino . 3. RFID.

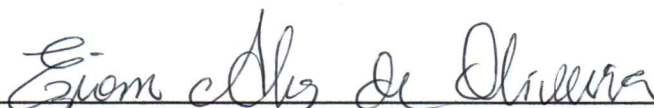
I. Título.

CDD: 005.4

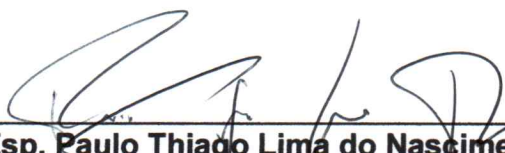
DANIELA NOGUEIRA FERRAZ CORNÉLIO

**Elaboração de um protótipo de ponto eletrônico
utilizando Arduino e RFID: Uma visão para o Instituto
Federal de Educação, Ciência e Tecnologia do Sertão
Pernambucano – Campus Floresta**

Monografia apresentada ao curso de Gestão de Tecnologia da Informação do Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano – Campus Floresta, como requisito parcial para obtenção do grau de Tecnólogo em Gestão de Tecnologia da Informação.



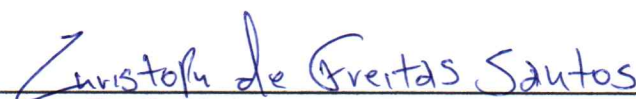
Profº. Eziom Alves Oliveira – Orientador
Instituto Federal do Sertão Pernambucano – Campus Ouricuri



Profº. Esp. Paulo Thiago Lima do Nascimento – Co-orientador
Instituto Federal do Sertão Pernambucano – Campus Floresta



Profº. Esp. Lincoln Tavares dos Santos – Avaliador
Instituto Federal de Pernambuco – Campus Igarassu



Profº. Christoph de Freitas Santos – Avaliador
Instituto Federal do Sertão Pernambucano – Campus Floresta

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer a Deus, por toda força e fé para seguir e alcançar meus objetivos e hoje poder chegar até aqui.

Gostaria de agradecer também aos meus pais e meu irmão, que sempre acreditaram muito em mim e na minha capacidade e determinação. Obrigada, com todo meu coração, por tudo. Apesar dos estresses, das brigas pelas noites não dormidas e pela rotina bagunçada, de qualquer forma sempre pude ter o apoio deles, como também de toda a minha família que sempre muito unida, apoia e dá força. Mais ainda o meu pai, que sempre acreditou em mim, mais do que eu mesmo, sonhou por mim. Pai, o melhor dessa conquista é te ver aí, me assistindo defender a minha monografia e concluir essa nossa tão sonhada etapa. E como o senhor mesmo diz, 'vem mais por aí'. Obrigada pai, por tudo!

Gostaria ainda de homenagear meu tio, eternamente lembrado, Mário. Seja onde estiver meu tio, meus sinceros agradecimentos, com carinho e amor. Só eu sei o quanto que eu sonhei em estar nessa etapa com o senhor presente, assistindo minha defesa. Muita saudade, muita mesmo. E deixo aqui meu carinho especial por tudo que o senhor representa em minha vida.

Bom gostaria de agradecer a uma pessoa, que não poderia deixar de citar aqui. É, você, Lary, minha grande companheira de todas as horas. Sempre vibrando com cada etapa vencida, com cada fase concluída. Estava lembrando que inúmeras vezes você falava que já me via apresentando o meu TCC, com toda empolgação e felicidade. Pois é, estou aqui hoje, e sem dúvida você tem uma parcela muito grande de ajuda nessa conquista. Um muito obrigada muito especial a você.

Não podendo esquecer os professores que jamais serão esquecidos, pois contribuirão de tal forma, que palavra nenhuma pode expressar. Pois é Eduardo Vergolino, 'missão dada é missão cumprida', muito obrigada viu? Todas as broncas, conselhos, livros, incentivo e credibilidade. 'Estamos cheios de problemas' né Lincoln Tavares? Mas estamos terminando esse, que sem dúvida, sem você não teria sido um caminho tão proveitoso e bom de seguir. Você sempre com as palavras certas, o ombro amigo, os conselhos, sempre um paizão. Obrigada por toda contribuição. Ao querido e admirável Wagner Pinheiro, que com certeza contribuiu positivamente nessa

minha trajetória, com uma inteligência e conhecimentos dignos de toda admiração. A Willma, Maria, Iran, Felipe, todos queridos demais, só tenho que agradecer por ter tido professores/amigos tão especiais, comprometidos e competentes. Vocês são inesquecíveis.

Meus queridos orientadores, Ezio, meu professor e amigo sempre, e Paulo Thiago, obrigada por entrarem nessa comigo, obrigada por acreditarem que conseguiríamos concluir. Apesar das muitas dificuldades encontradas no decorrer do caminho, espero atender as expectativas.

Gostaria de agradecer a todos os colegas de curso, que passamos juntos muitas coisas nesses anos. Em especial a dois verdadeiros amigos, Roama e Jairo, que construímos um laço muito lindo de amizade e carinho. Sem esquecer Christoph e Yulle, que sempre me ajudaram demais.

Sem contar em toda a equipe IF sertão campus Floresta, que desde que eu entrei me recebeu super bem, Sandra, Rafaela, Mayara, e todos os demais, que sempre que precisei estavam com toda boa vontade para ajudar e resolver. A galera de Química, que muitas vezes achavam que eu fazia era química e não GTI, né? Vocês são massa. Meus amigos, Taisinha, Welder, Batatinho, Rafa, enfim, cada um dos que sabem de alguma maneira contribuem na minha vida e moram no meu coração.

"Não deixe o barulho da opinião dos outros abafar sua voz interior. E mais importante, tenha a coragem de seguir seu coração e sua intuição. Eles de alguma forma já sabem o que você realmente quer se tornar. Tudo o mais é secundário."

Steve Jobs

RESUMO

É recorrente hoje o uso de protótipos para iniciar um projeto. É tido como positivo e útil em vários casos, então, surgiu a ideia da construção de um protótipo de um ponto eletrônico, sabendo também que pode-se adiantar informações sobre como o produto precisa funcionar, facilitando as tomadas de decisões sobre qual processo deve ser realizado, neste caso ainda, diminuir os possíveis erros e *bugs* do programa. O presente trabalho aborda a ideia de um ponto eletrônico, desenvolvido com um arduino, que é uma plataforma *open-source* de protótipo. Este *hardware* é capaz de simular e executar todas as funções desejadas para a criação da aplicação. Foi escolhido por sua flexibilidade de plataforma e na simplicidade de utilização, não deixando de lado o baixo custo na implementação. utilizando a tecnologia RFID (*radio frequency identification* – identificação por radiofrequência), que é uma tecnologia que utiliza a frequência de rádio para captura de dados, assim armazenando um número de série que identifique uma pessoa, um objeto ou outra informação em um microchip. Através dessa tecnologia é possível capturar de maneira automática dados de identificação com dispositivos eletrônicos, como por exemplo, as etiquetas eletrônicas, *tags*, *transponders*, que são basicamente, objetos que irão emitir um sinal de radiofrequência para que os leitores captem determinadas informações. Ao final do trabalho, teremos uma base de informações de como construir um equipamento tecnológico capaz de auxiliar na gestão de pessoas.

Palavras-chave: ponto eletrônico, arduino e RFID.

ABSTRACT

Prototypes usage aiming to set up projects is recurring nowadays. This usage is considered positive and useful in many situations. The idea of building a prototype of an electronic point, knowing also that it is possible to advance information about how the product needs to work, facilitating decision-making about which process should be done in this case also reduce possible errors and software bugs. This monograph discusses the idea of an electronic point, developed with Arduino, which is an open-source prototype platform. This hardware is able to simulate and perform all functions desired for the application requirements. Arduino was chosen for its flexible platform and simplicity of use, leaving aside the low cost implementation and using RFID (radio frequency identification), which is a technology that uses radio frequency to capture data, therefore, storing a serial number that identifies a person, an object or other information on a microchip. Through this technology it is possible to automatically capture and identify data with electronic devices, such as electronic labels, tags, transponders, which are basically objects that will emit a radio frequency signal so that readers are able to capture certain information. At the end of the work, we will have a base of information on how to build a technological equipment able to assist in human resources management.

Keywords: electronic point, Arduino and RFID.

LISTA DE ABREVIATURAS E SIGLAS

CTL	Conjunto de leis trabalhistas
MTE	Ministério do Trabalho e Emprego
SREP	Sistema de Registro Eletrônico
REP	Registrador Eletrônico de Ponto
ABREP	Associação Brasileira das Empresas Fabricantes de Equipamentos de Registro Eletrônico de Ponto
RFID	<i>Radio Frequency Identification</i> – Identificação por Radiofrequencia
IFF	<i>Friend-or-FOe</i> – Sistema de Identificação de Amigos ou Inimigos
UHF	Ultra alta frequência
RF	Radiofrequencia
ROM	Read Only Memory
TCP	<i>Transmission Control Protocol</i>
UDP	<i>User Datagram Protocol</i>
DC	Corrente contínua
SGBD	Sistema de Gerenciamento de Banco de Dados
GNU	<i>General Public License</i>
TI	Tecnologia da Informação
SO	Sistema Operacional
JDBC	<i>Javs Database Connectivity</i>
SQL	<i>Structure Query Language</i>
UML	<i>Unified Modeling Language</i> ou Linguagem Unificada de Modelagem

LISTA DE FIGURAS

FIGURA 1 - CONTROLE MECÂNICO	19
FIGURA 2 - SISTEMA IFF	25
FIGURA 3 - FUNCIONAMENTO DO RFIDV	26
FIGURA 4 - ETIQUETAS EM FORMATOS DIFERENTES.....	29
FIGURA 5 - ETIQUETAS DIVERSAS	30
FIGURA 6 - CLASSES DAS ETIQUETAS.....	30
FIGURA 7 - RFID	32
FIGURA 8 - MODELOS DE LEITORES	33
FIGURA 9 - LEITOR RFID PARA ARDUINO	33
FIGURA 10 - FAIXAS DE FREQUÊNCIA.....	34
FIGURA 11 - PLACA ATMEGA328	37
FIGURA 12- PINAGEM PLACA ARDUINO	41
FIGURA 13 - CHIP ATMEGA328	42
FIGURA 14 - PINAGEM DO MODULO RFID.....	43
FIGURA 15- DIAGRAMA DE CLASSES.....	48
FIGURA 16 - DIAGRAMA DO FUNCIONAMENTO DO PROJETO	50
FIGURA 17 - PROTOTIPO: PONTO ELETRÔNICO ARDUINO	52
FIGURA 18 - TELA DO NETBEANS: COMUNICAÇÃO SERIAL	52
FIGURA 19 - PERSONALIZAÇÃO DO CARTÃO ELETRONICO.....	53

LISTA DE TABELAS

TABELA 1 - DADOS TÉCNICOS 38

SUMÁRIO

1. INTRODUÇÃO	15
2. JUSTIFICATIVA	17
3. OBJETIVO	18
3.1. Geral	18
3.2. Específicos	18
4. REFERENCIAL TEÓRICO	19
4.1. Sistema de Ponto Eletrônico	19
4.2. Identificação por Radiofrequência - RFID	24
4.2.1. Etiquetas ou Tag	27
4.2.1.1. Tag Passivas	27
4.2.1.2. Tag Ativas	28
4.2.2 Formas	29
4.2.3. Classes das etiquetas	30
4.2.4. Antenas	31
4.2.5. Leitores	32
4.2.6. Frequência de Operação	34
4.3. Arduino	36
4.3.1. Pinagem	41
4.4. Banco de Dados	44
4.4.1. Sistema de Gerenciamento de Banco de Dados – SGBD	44
4.4.2. MySQL	45
4.4.3. UML	46
4.4.3.1. Diagrama de Classe	47
4.5. Uso da tecnologia da gestão empresarial	49
5. METODOLOGIA	50

6. RESULTADOS OBTIDOS 52

7. CONSIDERAÇÕES FINAIS 54

7.1. Considerações gerais 54

7.2. Considerações sobre o *hardware* 54

7.3. Considerações sobre o *software* 54

7.4. Trabalhos futuros 54

8. REFERÊNCIAS..... 55

APÊNDICES..... 60

APENDICE A – Código para controle de porta: 60

APÊNDICE B – Código para a comunicação serial 65

APÊNDICE C – Conexão JDBC 67

APÊNDECE D – Código para o controle de Acesso – RFID – ARDUINO 72

1. INTRODUÇÃO

As leis são constantemente modificadas no Brasil, pois é comum nos depararmos nessas alterações com situações que dificultam a sua execução, isto acontece a fim de buscar melhores meios de garantir a integridade da informação. No caso da legislação trabalhista o cenário não é diferente, vem sendo criados métodos que buscam assegurar ao trabalhador e as empresas quanto seus direitos e deveres relacionados ao trabalho.

A alteração mais recente fica por conta da nova portaria do Cartão de Ponto, do Ministério do Trabalho, esta vem sendo vista como um instrumento de suporte que visa melhorar o controle dos horários dos funcionários, podendo assim acompanhar mais precisamente o cumprimento da jornada de trabalho, assegurando os direitos referentes a esse assunto, dentre outras vantagens, porém, surgem questionamentos negativos relativos ao custo de adquirir tal equipamento e um aumento na logística da sua utilização.

Um dos aspectos primordiais neste desenvolvimento está na utilização de uma ferramenta de baixo custo que atenda as necessidades e exigências para tal aplicação. Pensando nisto, tem-se a placa de desenvolvimento Arduino, que é uma ferramenta “*open source*”, que como o nome já diz, utiliza códigos abertos. Dentre as suas características, além do baixo custo, tem-se a simplicidade e a possibilidade de utilizá-lo em diversos sistemas operacionais, como o Windows, Macintosh IOS e Linux.

A proposta deste trabalho é a realização do desenvolvimento de um protótipo de Ponto Eletrônico utilizando a plataforma de prototipagem Arduino, que é de código aberto, para o Instituto Federal do Sertão Pernambucano (IF Sertão) Campus Floresta. O trabalho foi estruturado da seguinte maneira: o capítulo 1 terá a introdução sobre a temática, ficando para o capítulo 2 a justificativa, no capítulo 3 os objetivos geral e específico, no capítulo 4 será o referencial teórico, onde será realizada uma revisão bibliográfica sobre temas como: Sistema de Ponto Eletrônico, Arduino, Banco de Dados que irá fundamentar o trabalho. No capítulo 5 estará a metodologia usada no desenvolvimento. No capítulo 6 trará a apresentação do protótipo desenvolvido

apresentando ainda a execução do mesmo através dos resultados obtidos e por fim no capítulo 7 a conclusão e as propostas para trabalhos futuros.

2. JUSTIFICATIVA

Este trabalho é justificado pela necessidade de criação de um projeto que atenda uma demanda reprimida do campus Floresta, que é a inexistência de um ponto eletrônico para os servidores da instituição. Vendo também que seria uma ferramenta bastante eficiente para a Gestão de Pessoas, facilitando o controle das informações importantes quanto ao cumprimento de carga horária de trabalho. A folha de ponto usada atualmente requer muito trabalho para o cálculo das horas trabalhadas no mês, esse cálculo é executado manualmente pelo coordenador do curso. O interesse de pesquisar sobre temas de interesse coletivo. Além disso, no processo de elaboração deste trabalho serão utilizados vários conhecimentos que foram adquiridos durante a permanência no curso de Gestão da Tecnologia da Informação (GTI).

No IF Sertão campus Floresta não existe, atualmente, em funcionamento nenhum sistema de Ponto Eletrônico, ainda existe uma necessidade para que o Gestor da Instituição obtenha informações mais imediatas e precisas. Este protótipo teria como função auxiliar a equipe de Gestão de Pessoas do campus para coordenar melhor os servidores e assegurando que tais informações fiquem armazenadas e não sofreram nenhum dano ou alteração.

Conforme dito, por Gustavo Setúbal (2013), “a área de tecnologia da empresa busca cada vez mais estar qualificada em técnicas e ferramentas a auxiliar a suas atividades.” Ainda segundo ele, é importante acompanhar as tendências do mercado, buscando proporcionar facilidades e qualificada nas técnicas e ferramentas que auxiliem nas suas atividades organizacionais.

3. OBJETIVO

3.1. Geral

Construção de um protótipo de um Ponto Eletrônico utilizando arduino e RFID, tendo uma perspectiva de criação para utilização no Campus Floresta.

3.2. Específicos

- Pesquisar teoricamente sobre o ponto eletrônico, arduino e RFID;
- Apresentar características da tecnologia utilizada na construção do protótipo;
- Desenvolver armazenamento das informações coletadas;

4. REFERENCIAL TEÓRICO

4.1. Sistema de Ponto Eletrônico

O Ponto Eletrônico surgiu em 1888, denominado inicialmente como Relógio de Ponto, sendo criado por Willard Bundy, nos Estados Unidos. No Brasil, apenas em 1930 que começou a ter relação com a classe trabalhadora e empresarial, a fim de atender necessidades advindas do controle da jornada de trabalho. Nesse período existia um decreto sancionado pelo presidente da época Getúlio Vargas, a respeito de um conjunto de leis trabalhistas (CLT), em que determinava três maneiras para controlar a jornada de trabalho: através do ponto manual, o controle mecânico ou do ponto eletrônico.

A princípio só existia o ponto manual e o controle mecânico (Figura 1), em que era feito através de um livro de ponto onde os funcionários escreviam os horários ou pela marcação em um relógio mecânico.

Figura 1 - Controle Mecânico



Fonte: <http://anos80amelhorepoca.blogspot.com.br/2016/08/maquina-de-bater-ponto.html> (2016)

Apenas a partir da década de 80 que surgiu o ponto eletrônico, sendo oficialmente reconhecido através da Lei 7.855 de 24 de Outubro de 1989. Conforme cita no § 2º, “para os estabelecimentos de mais de dez trabalhadores será obrigatória a anotação da hora de entrada e de saída, em registro manual, mecânico ou eletrônico, conforme instruções a serem expedidas pelo Ministério do Trabalho”.

Então o Ministério do Trabalho e Emprego (MTE), pensando em assegurar melhor os trabalhadores e o empregador aprovou a Portaria Ministro de Estado do Trabalho e Emprego de nº 1.510 de 21 de Agosto de 2009, no seu artigo primeiro diz que se deve “disciplinar o registro eletrônico de ponto e a utilização do Sistema de Registro Eletrônico de Ponto – SREP”. E no paragrafo único conceitua o SREP como sendo um:

Conjunto de equipamentos e programas informatizados destinado à anotação por meio eletrônico da entrada e saída dos trabalhadores das empresas, previsto no art. 74 da Consolidação das Leis do Trabalho - CLT, aprovada pelo Decreto-Lei nº 5.452, de 1º de maio de 1943. (MTE, 2009)

O sistema busca registrar e armazenar fielmente as informações quantos aos horários marcados, não podendo haver nenhuma ação que desvirtue a finalidade do ponto. Possuindo o Registrador Eletrônico de Ponto (REP), que segundo a mesma portaria citada anteriormente, afirma que o REP é um “equipamento de automação utilizado exclusivamente para o registro de jornada de trabalho e com capacidade para emitir documentos fiscais e realizar controles de natureza fiscal, referentes à entrada e à saída de empregados nos locais de trabalho.” é percebido que para utilização do SREP é necessário utilizar outros meios de registro, sendo então obrigatoriamente utilizado o REP.

Esse registrador deverá contar com alguns requisitos, como citados no artigo 4º:

I - relógio interno de tempo real com precisão mínima de um minuto por ano com capacidade de funcionamento ininterrupto por um período mínimo de mil quatrocentos e quarenta horas na ausência de energia elétrica de alimentação;

II - mostrador do relógio de tempo real contendo hora, minutos e segundos;

III - dispor de mecanismo impressor em bobina de papel, integrado e de uso exclusivo do equipamento, que permita impressões com durabilidade mínima de cinco anos;

IV - meio de armazenamento permanente, denominado Memória de Registro de Ponto - MRP, onde os dados armazenados não possam ser apagados ou alterados, direta ou indiretamente;

V - meio de armazenamento, denominado Memória de Trabalho - MT, onde ficarão armazenados os dados necessários à operação do REP;

VI - porta padrão USB externa, denominada Porta Fiscal, para pronta captura dos dados armazenados na MRP pelo Auditor- Fiscal do Trabalho;

VII - para a função de marcação de ponto, o REP não deverá depender de qualquer conexão com outro equipamento externo; e

VIII - a marcação de ponto ficará interrompida quando for feita qualquer operação que exija a comunicação do REP com qualquer outro equipamento, seja para carga ou leitura de dados. (MTE, 2009)

Essa portaria também prevê, no seu artigo 6º, que algumas operações deverão ser gravados de forma permanente no MTE, como:

I - inclusão ou alteração das informações do empregador na MT, contendo os seguintes dados: data e hora da inclusão ou alteração; tipo de operação; tipo de identificador do empregador, CNPJ ou CPF; identificador do empregador; CEI, caso exista; razão social; e local da prestação do serviço;

II - marcação de ponto, com os seguintes dados: número do PIS, data e hora da marcação;

III - ajuste do relógio interno, contendo os seguintes dados: data antes do ajuste, hora antes do ajuste, data ajustada, hora ajustada; e

IV - inserção, alteração e exclusão de dados do empregado na MT, contendo: data e hora da operação, tipo de operação, número do PIS e nome do empregado.

Parágrafo único. Cada registro gravado na MRP deve conter Número Sequencial de Registro - NSR consistindo em numeração sequencial em incrementos unitários, iniciando-se em 1 na primeira operação do REP (MTE, 2009).

Existiam opiniões contrárias, porém, tem recebido aval positivo da maioria dos envolvidos. De acordo com a ABREP – Associação Brasileira das Empresas Fabricantes de Equipamentos de Registro Eletrônico de Ponto (2016):

“Os resultados da pesquisa de satisfação do uso do REP realizada recentemente pelo Instituto AGP, mostram que 74% dos empregadores e 78% dos profissionais aprovaram a novidade. A percepção é a de que o novo relógio de ponto melhorou a relação trabalhista e está oferecendo mais segurança a empregados e empregadores.” (ABREP, 2016)

Tanto para as empresas como o empregados, alegam que o novo sistema melhorou a relação trabalhista ao proporcionar mais segurança e transparência entre as partes. De acordo com um estudo realizado por Pimenta (2011), 60% dos profissionais e 70% das empresas se sentem mais protegidos com o novo registro de ponto. Além disso, o relacionamento e a confiança entre ambos melhorou 59%, contra 14% que não acreditam nessa melhora. O documento indica ainda que após a troca de processo, o volume de questionamentos relacionados ao pagamento da hora extra diminuiu em 28% dos casos.

Nos novos sistemas de ponto eletrônico, vem em seu registro de dados informações mais aperfeiçoadas a respeito da identificação do colaborador, sem contar com outras características, como ser inviolável e garantir mais segurança na transmissão e registro das informações. Diferente dos mais antigos que não contavam com tantas tecnologias e segurança. Então, a medida que essa tecnologia vai crescendo, os pontos eletrônicos vão ficando mais sofisticados no quesito operacional.

Existem alguns os modelos disponíveis atualmente no mercado, um que funciona por meio de passagem ou aproximação do crachá e o que funciona por biometria. No caso da biometria é mais seguro e confiável, em relação a identificação do usuário, podendo variar a forma de identificação, indo desde os mais convencionais que utilizam informações provenientes da impressão digital, topografia facial que identifica o relevo do rosto, estrutura da íris, desenho das mãos, mapa das veias sanguíneas, análise da voz e reconhecimento da grafia, como também maneiras mais avançadas, como por exemplo, pela forma de caminhar, odor, dentre outras.

Assim, observa-se que com as tecnologias existentes a utilização do ponto eletrônico vai ganhando mais confiança, funcionalidades e formas de utilização. Tornando-se mais seguro e preciso.

4.2. Identificação por Radiofrequência - RFID

A tecnologia Rádio Frequency Identification (RFID) é uma tecnologia que utiliza a frequência de rádios ou variações de campo magnético para comunicação entre componentes eletrônicos e sua identificação (Barge et al, 2013).

O RFID consiste em um sistema de envio de dados de radiofrequência de um dispositivo móvel para um leitor, utilizando tecnologias sem fio. Sendo composto basicamente por um leitor com antena, chamado *transceiver*, etiquetas ou *tag*, chamado *transponder* e o computador.

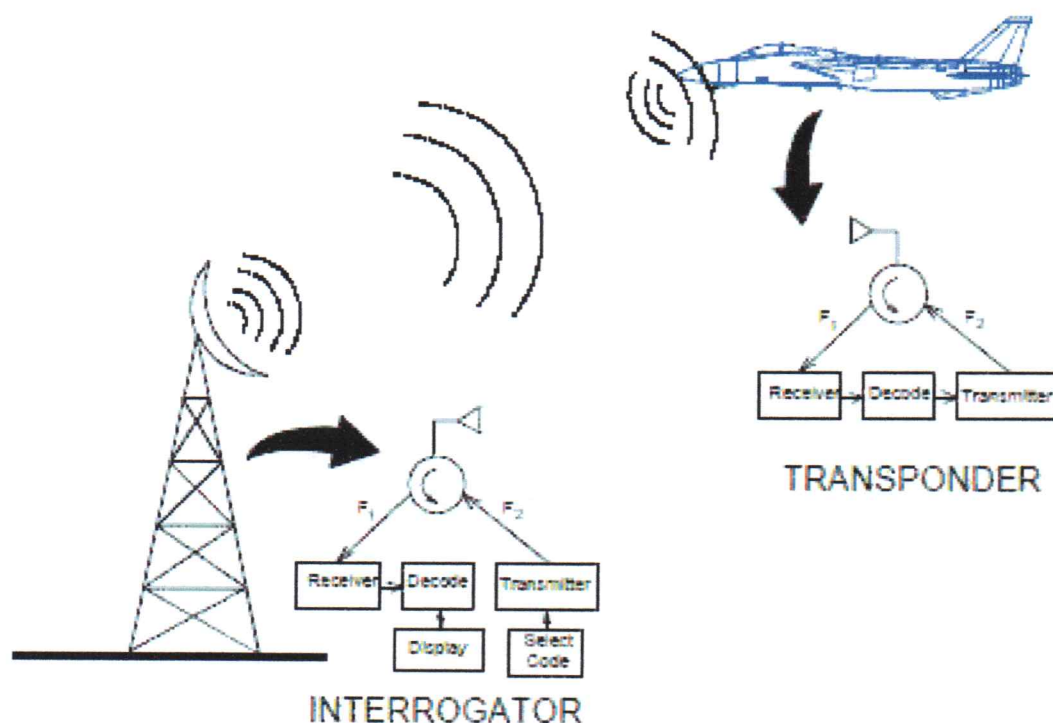
Conforme dito por Dias (2008):

RFID (*Radio Frequency Identification* ou Identificação por Radiofrequência) é um método que utiliza ondas eletromagnéticas para acessar dados armazenados em um microchip acoplado a uma pequena antena identificando automaticamente os objetos nele fixado.

Segundo Glover & Bhatt (2007), apesar do sistema de identificação baseado na radiofrequência ou a tecnologia RFID, está adquirindo uma dimensão cada vez maior no mercado atual, desde os anos 30 já vem sendo utilizado, quando o exército e a marinha precisavam identificar adequadamente os veículos que se aproximavam das duas bases, surgindo então o sistema *Identification Friend-or-FOe* (IFF – Sistema de Identificação de Amigos ou Inimigos), permitindo que existisse uma distinção entre os aliados e os inimigos que se aproximavam.

Nos primeiros sistemas IFF, o interrogador era o radar e o *transponder* era uma caixa grande de tubos com *dials* e interruptores, os quais eram os componentes mais importantes. Funcionando basicamente como visto na figura 2: a estação no solo enviava um sinal de radar e o *transponder* quando recebia esse sinal o refletia de volta, fazendo assim que o radar recebesse um retorno mais forte, por causa dessa reflexão, se o retorno fosse forte era identificado que o veículo que estava por chegar era um aliado (GLOVER & BHATT, 2007 apud DIAS, 2008, p. 2-3).

Figura 2 - Sistema IFF



Fonte: <http://www.rfcafe.com/references/electrical/ew-radar-handbook/iff-identification-friend-or-foe.htm> (2016)

Outro sistema utilizando a tecnologia RFID, foi desenvolvido no início dos anos 90, o UHF (Ultra alta frequência), o qual permitiria uma maior velocidade na transmissão de dados ao mesmo tempo que permitiria que a leitura desses dados fossem feitas a uma distância consideravelmente maior (RFIDJOURNAL, 2008 apud DIAS, 2008, p. 3).

Conforme dito por Dias (2008), a tecnologia de RFID precisa de três elementos para seu funcionamento, que são: Etiqueta ou *tag*, Leitor e Antenas. Em que, normalmente, o leitor está conectado a um computador, a fim de possibilitar a realização do processamento e da leitura dos dados das etiquetas.

O princípio de funcionamento do RFID é simples, funciona com o leitor emitindo um sinal de radiofrequência através da antena, podendo estar acoplada ao leitor ou ligada por um fio ao mesmo, com o intuito de localizar as tags (ou etiquetas) para captarem e emitir o sinal constantemente quando está na área coberta pelo leitor. Assim, essa tag responde ao leitor da mesma maneira, passando todas as informações do produto contidas na etiqueta. Então, o leitor envia, em tempo real, essas informações recebidas para o sistema computacional no qual tem instalado um

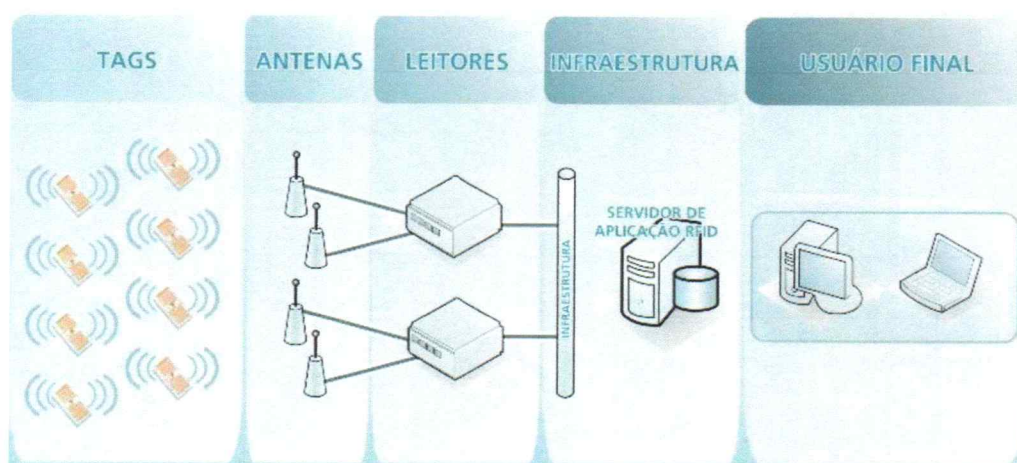
software específico para reconhecer e identificar essas informações para conseguir fazer o gerenciamento convergindo para o bom funcionamento deste sistema.

Para melhor explicar sobre o funcionamento do RFID, Dias (2008) diz:

O sistema RFID possui o seguinte funcionamento: um aparelho com função de leitura envia, por meio de uma antena, sinais de radiofrequência em busca de objetos a serem identificados. No momento em que um dos objetos é atingido pela radiação, ocorre um acoplamento entre ele e a antena, o que possibilita que os dados armazenados no objeto sejam recebidos pelo leitor. Este trata a informação recebida (identificação) e a envia ao computador.

A seguir, a figura 3 explica como é o funcionamento do RFID, em que a antena emite um campo RF, que é recebido pela *tag*, que por sua vez responde ao sinal enviando suas informações para o leitor, assim, o leitor envia essas informações para o sistema.

Figura 3 - Funcionamento do RFIDV



Fonte: <http://www.activa-id.com.br/o-que-e-rfid#.V84ISpgrLIU> (2016)

De acordo com o *RFID Journal Brasil* (2011), milhares de empresas ao redor do mundo utilizam RFID hoje para melhorar a eficiência interna. Segundo Bernardo (2004), especialistas em novidades da tecnologia afirmam que mais cedo ou mais tarde as etiquetas RFID estarão nos produtos que qualquer consumidor que vier a comprar e que esses pequenos chips revolucionaram a logística de estoque, o rastreamento e o gerenciamento de todo o processo. Abaixo algumas vantagens da utilização do RFID:

- Realizar leitura sem a necessidade de uma grande proximidade com o leitor;

- Prevenção de roubos e falsificação de mercadoria;
- Verificação de validade de produtos;
- Precisão e velocidade de armazenamento de informações.

As etiquetas ou *tag* RFID podem ser lidas a grandes distâncias, sem a necessariamente está no mesmo local, porém, estejam dentro do alcance da antena do leitor, para que assim seja possível efetuar as leituras. Além disso, essas *tags* são resistentes e possuem uma alta durabilidade, capaz de serem usadas em ambientes hostis, ou na parte interna dos objetos, pois as ondas de rádio podem ultrapassar diferentes tipos de materiais.

Outro grande diferencial é a possibilidade de leitura e escrita, assim permitindo que seja realizada uma gravação, edição e até apagar constantemente as informações. Além da sua elevada capacidade de memória, a qual propicia o armazenamento de todas as informações relacionadas. O fato de ser regravável e a sua alta durabilidade, torna a etiqueta RFID um material reutilizados diversas vezes dependendo da aplicação. (BERNARDO, 2004; FIGUEREIDO, 2004; STANTON, 2004; SOUZA *et al.*, 2009)

4.2.1. Etiquetas ou Tag

“As etiquetas de RFID possuem dois elementos básicos: um *chip* e uma antena, o *chip* e a antena são montados para formar o *inlay*.” (INTERMEC, 2007, p. 2).

As *tags* são classificadas em duas grandes categorias, sendo uma passivas e outra ativas, que são classificadas em classes, possuindo encapsulamento de diferentes formatos.

4.2.1.1. Tag Passivas

As *Tags* passivas não possuem a sua própria fonte de energia, para que elas possam interagir com os leitores é necessário que sejam energizadas. Essa fonte de

alimentação é fornecida no sistema RFID através do leitor por meio das antenas, que irradia energia através de um campo magnético, fazendo assim com que a etiqueta interaja (FINKENZELLER, 2010 apud MARQUES, 2012, p. 27).

O campo eletromagnético produzido pelo leitor que estimula a etiqueta é denominado de *Near Field*, gerando uma corrente elétrica reduzida, para que a *tag* passiva possa emitir um pulso de resposta de volta para o leitor com as informações contidas no seu circuito integrado. (THORNTON, 2006 apud MARQUES, 2012).

Conforme foi citado por Dalfovo e Hostins (2010), com relação as etiquetas passivas, diz que elas possuem normalmente memórias ROM (*Read Only Memory*), que são as memórias somente de leitura, as quais tendem variar de 32bits a 128Kbits, essas informações não podem ser modificadas. Por não possuírem grande poder computacional as *tags* passivas possuem menor custo.

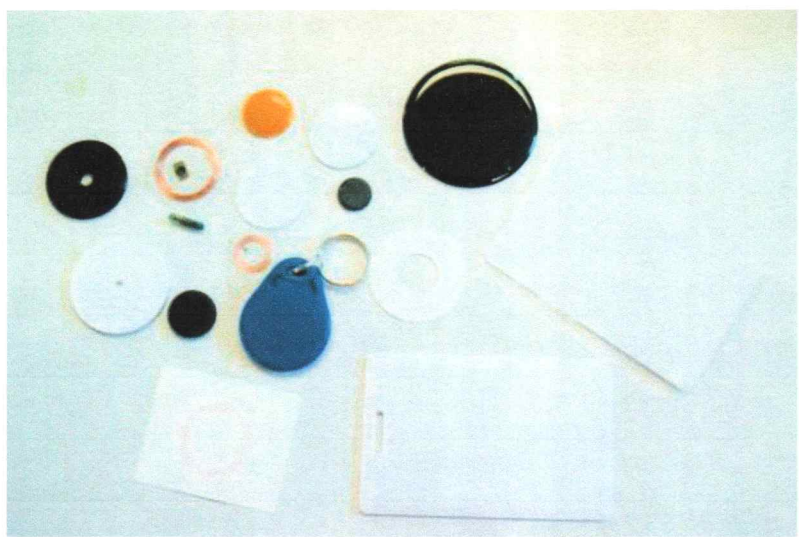
4.2.1.2. Tag Ativas

Tag ativas possuem sua própria bateria para alimentação, responsável por aumentar sua capacidade de armazenamento e processamento, podendo ler e gravar informações do ativo, por esse motivo, elas são mais caras que as etiquetas passivas. (ROBERTS, 2006 apud MARQUES, 2012, p. 28).

Rosa (2006) diz que as etiquetas ativas podem operar em um faixa de 455 Mhz e 5,8 Ghz, além disso possibilitar a comunicação com os leitores em distâncias maiores que as *tags* passivas, essas *tags* ativas possuem uma vida útil de mais de 7 anos.

Existe também uma subcategoria de *tags*, conhecidas como *tags* semi-passivas. As quais consomem menos energia em relação as ativas, e quando comparada com as passivas, seu alcance de leitura é maior, porém por possuir bateria o seu custo é mais elevado que as passivas. (PRERADOVIC e KARMARKAR, 2006 apud MARQUES, 2012)

Figura 5 - Etiquetas diversas

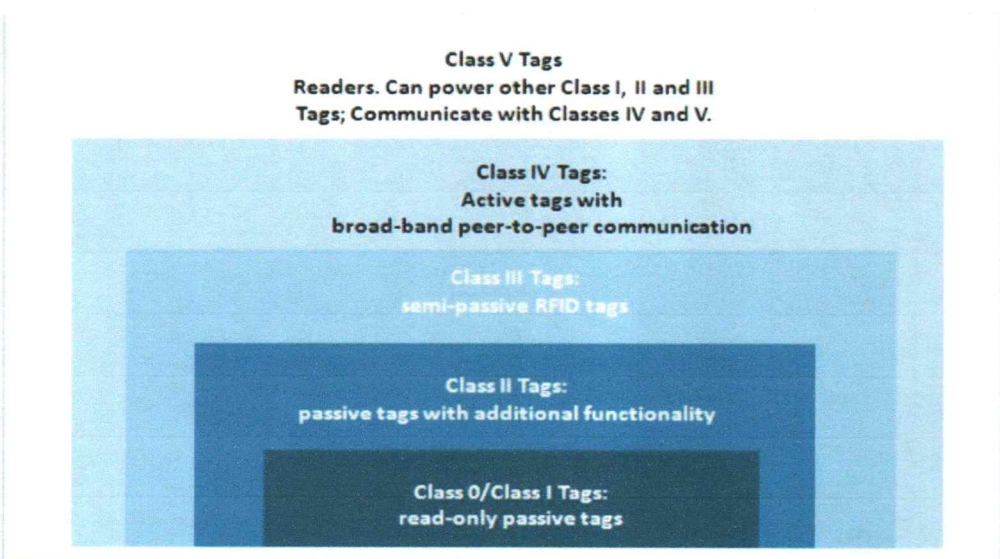


Fonte: <http://www.rfidbr.com.br/index.php/tags-rfid.html> (2016)

4.2.3. Classes das etiquetas

Conforme Dalfovo e Hostins (2010), existem seis classes de etiquetas que variam da 0 até a V (5), sendo cada *tag* classificada em uma dessas classes variando de acordo com suas funcionalidades, conforme mostrado na figura 6.

Figura 6 - Classes das etiquetas



Fonte: www.centrenational-rfid.com (2016)

Que de acordo com o site CNRFID - Centro Nacional de RFID (2016), a classe 0 a I são as etiquetas passivas, que fazem apenas somente leitura, e só permite ler a identificação de apenas uma tag. Já na classe II, as etiquetas ~~passivas~~ para funções adicionais, sendo escrita na memória. A classe III são as etiquetas passivas assistida por bateria. A classe IV, são as tags ativas, com comunicações de banda larga, funcionando como "~~peer-to-peer~~", a P2P, ou ponto-a-ponto. E por fim, a classe V, que são as interrogantes, pois alimenta as etiquetas de classe 0, I, II e III, e comunica-se com a classe IV.

Então Prado, Pereira e Politano (2006) explicam que a classe 0 e 1 são destinadas as *tags* passivas, a diferença é que a classe 0 possui um protocolo de leitura distinto da classe 1, essas categorias permitem que a informação seja gravada apenas uma vez. Depois da gravação, será possível apenas fazer a leitura do que foi gravado anteriormente.

A classe 2 são as *tags* passivas, porém apresentam funcionalidades adicionais das anteriores. A classe 3 é destinada às *tags* semi-passivas, essas já possuem baterias próprias e suportam leituras a maiores distâncias. A ~~classe~~ 4 são etiquetas ativas e apresentam autonomia para iniciar as suas comunicações. E por fim a classe 5 que possui todas as características das classes anteriores e ainda pode fornecer energia para outras *tags* e se comunicar com outros leitores (PRADO, PEREIRA e POLITANO, 2006).

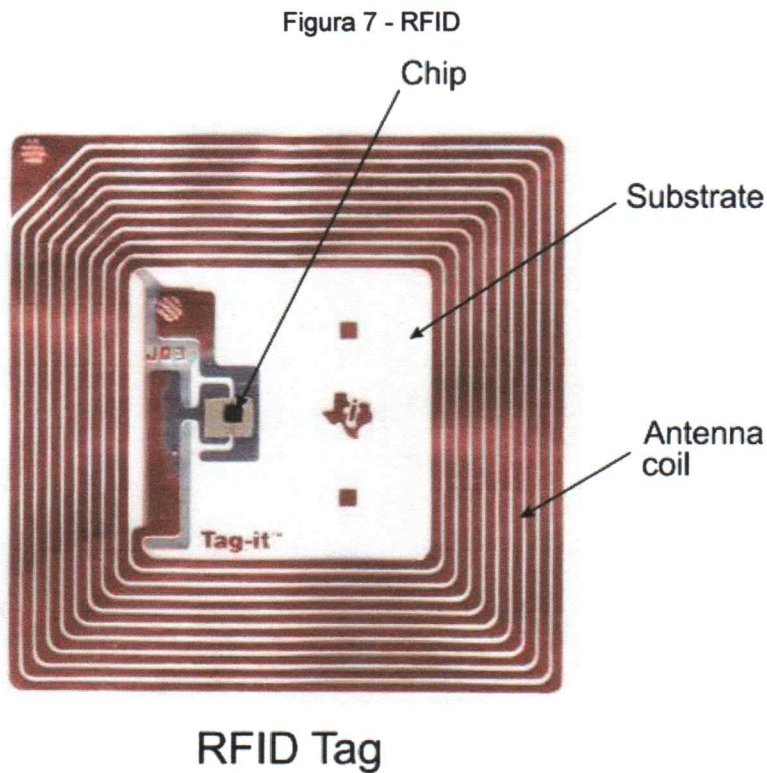
4.2.4. Antenas

Conforme dito por Dalfovo & Tamborlin (2010) as antenas são fabricadas em diversos tamanhos e formatos, havendo configurações e características distintas, sendo cada uma para um tipo de aplicação.

A antena é responsável pela ativação da *tag*, através de sinais de rádio, realizando a troca de informações com essas etiquetas. (BERNARDO, 2004).

Antunes, Werner e Alves (2011) dizem que as informações recebidas pela *tag* são capturadas pela ~~ante~~na, que compõe essa etiqueta, e é responsável pela ativação da *tag*. Após realizar essa ativação é possível estabelecer a comunicação com os leitores.

O RFIDJournal (2011) esclarece sobre as antenas, que elas estão presentes nas *tags* passivas como visto na figura 7 costumam ter um formato em espiral e as etiquetas ativas podem possuir diversas formas de antenas.



Fonte: <http://endtimetruth.com/rfid/> (2016).

4.2.5. Leitores

Para Dias (2008) os leitores têm o objetivo de criar, amplificar sinais de radiofrequência, principalmente para as tags passivas, e enviar os sinais para as antenas. Além disso são responsáveis por receber as respostas das etiquetas, organizar os dados e armazenar em um computador. Além disso, Dias (2008), diz que para que as informações fornecidas pelas etiquetas inteligentes possam ser entendidas pelo usuário, o leitor deve se comunicar com um software.

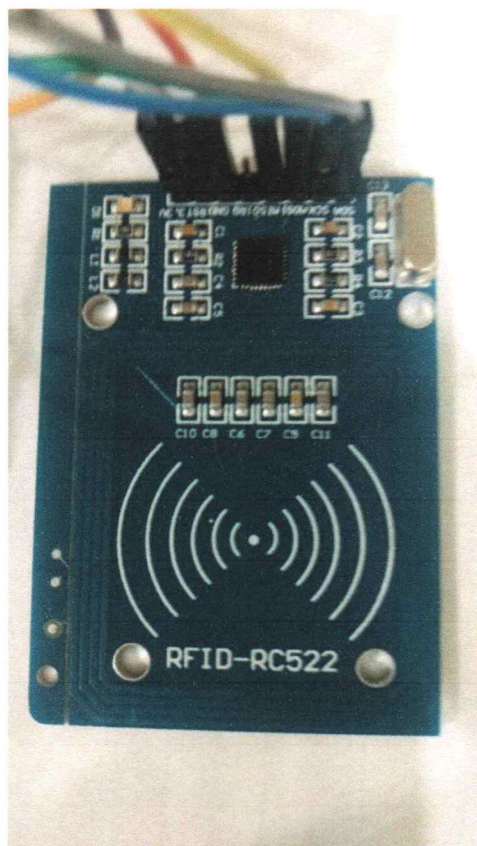
Os leitores variam de tamanho, estes podem chegar a ser muito pequeno ou possuir o tamanho de computador, podem ser fixados em paredes, ou até mesmo pertencer a um dispositivo móvel (BHATT & GLOVER, 2007).

Figura 8 - Modelos de Leitores



Fonte: <http://saladaautomacao.com.br/leitores-de-rfid/> (2016)

Figura 9 - Leitor RFID para Arduino



Fonte: O autor (2016)

Bhatt & Glover (2007) afirmam que os leitores precisam utilizar Radiofrequência - RF para se comunicar com as *tags*, sendo assim, os leitores precisam conter pelo

menos uma antena para fazer tal comunicação com as etiquetas. Como o leitor precisa se comunicar com um servidor ou algum dispositivo da rede, torna-se necessário que exista alguma interface de rede, como por exemplo o RJ45. Existem alguns leitores que possuem comunicação *bluetooth* ou comunicação sem fio.

4.2.6. Frequência de Operação

Conforme dito por Bernardo (2004) os Sistemas de Identificação por Radiofrequência têm faixas de operação e define a taxa de transferência entre a *tag* e o leitor e a distância de leitura. Os sistemas de baixa frequência vão de 100Khz a 500Khz, esses são utilizados em casos de leituras a uma distância curta, possuindo então um custo operacional menor. Na figura 10 é possível observar as faixas de frequência, suas características e suas aplicações.

Figura 10 - Faixas de Frequência

Banda de Frequência	Características	Aplicações Típicas
Baixa: 100 a 500 KHz	- Faixa de curta até média leitura - Baixo custo - Baixa velocidade de leitura	- Controle de acesso - Identificação de animal - Controle de inventário
Média: 10 a 15 MHz (também denominada Alta)	- Faixa de curta até média leitura - Potencialmente de baixo custo - Média velocidade de leitura	- Controle de acesso - Smart cards
Alta: 850 a 950 MHz e 2,4 a 5,8 GHz (também denominada Ultra Alta)	- Faixa larga de leitura - Alta velocidade de leitura - Alto custo - Linha de visão requerida	- Monitoração de veículos em estradas

Fonte: www.teleco.com.br

Os sistemas de alta frequência possuem um faixa de 850Mhz a 950Mhz e de 2,4Ghz a 5,8Ghz, e são utilizados para longas distâncias, e com isso o custo operacional é mais elevado sendo utilizados para localizar e rastrear veículos e para coleta automática de dados (BERNARDO, 2004).

Para esse projeto de protótipo, a utilização de *tags* de baixa frequência poderá ser a mais adequada, pois essa faixa é mais apropriada para utilização em controles de acesso, sendo assim, é satisfatório para o Ponto Eletrônico.

4.3. Arduino

Segundo informações do site, Arduino (ARDUINO, 2012) “é uma plataforma open-source de prototipagem eletrônica baseada na flexibilidade, hardware e software fácil de usar”. Ou seja, baseada em hardware livre e software colaborativo. Foi desenvolvida para ser completamente autônoma, precisando apenas de uma fonte de alimentação que coloque em execução duas rotinas. Foi feito para entusiastas, artistas, designers, programadores e qualquer um que esteja interessado em criar objetos e/ou ambientes interativos. Ainda segundo o site da Arduino (2012) os fundadores do Arduino são: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, e David Mellis.

Conforme citado por Oliveira e Carvalho (2014), o Arduino é também um microcontrolador, conforme explicado na seção acima, que nos permite criar uma grande variedade de projetos. O movimento DIY (*do it yourself* - “faça você mesmo”, em tradução literal), conforme explica Chris Anderson em seu livro “Makers: A nova revolução industrial” se apropriou do Arduino por conta de seu baixo custo e grande gama de possibilidades, criando desde Drones até próteses robóticas.

Segundo Werneck (2009) a plataforma Arduino busca atingir um número de pessoas, entusiastas de eletrônica ou hobbystas para desenvolver projetos com qualidade, porém de forma fácil e de valor acessível. Essa plataforma auxilia profissionais em programação que não possuem conhecimento em eletrônica, pois podem realizar experimentos em um equipamento prático. Então, essa plataforma atinge os leigos que tem o desejo de elaborar experimentos funcionais de alto nível, partindo do zero.

De acordo com Werneck (2009), existem mais plataformas de microcontroladores criadas com o mesmo objetivo, mas não mantendo o foco em características como facilidade de uso e preço acessível como o Arduino.

Arduino é uma plataforma de computação física Open-source elaborado em uma placa com entradas, saídas e um ambiente de desenvolvimento que implementa a linguagem Processing/Wiring. É um ambiente de desenvolvimento multiplataforma

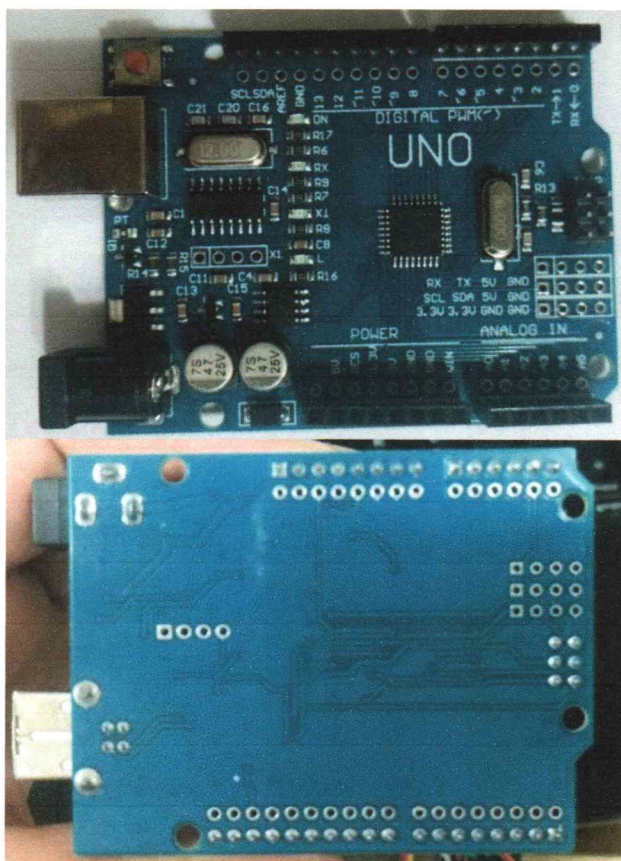
podendo criar as rotinas para Windows, Linux e Mac OS X. Sua linguagem de programação se assemelha com C++ e JAVA, porém o Arduino segue uma forma simplificada (Arduino, 2016).

Um dos fatores que assegurou o sucesso do Arduino deve-se ao valor acessível em adquirir uma placa pronta, permitindo que os admiradores da robótica ingressassem nessa área de forma facilitada não desembolsando valores exorbitantes como outras soluções existentes no mercado (Rodrigo, 2011).

Abaixo segue a plataforma Arduino utilizada no trabalho:

- Arduino Uno: Placa baseada no ATmega328(figura 11), que pode receber sinais de vários sensores eletrônicos e lidar com essas informações para controlar motores, luzer, servos e qualquer outro atuador.

Figura 11 - Placa ATmega328



Fonte: O autor

Na tabela 01 estão os dados técnicos do Arduino fornecidos pelo fabricante.

Tabela 1 - Dados Técnicos

Tensão de operação	5V
Tensão de entrada (recomendada)	7 – 12V
Tensão de entrada (limites)	6 – 20 V
Pinos digitais	14 (6 com saída PWM)
Pinos analógicos	6
Clock	16 MHz

Como mostrado no site Microchip Atmel (2016), o microcontrolador utilizado no Arduino é o ATmega328 da Atmel, que utiliza a arquitetura de Havard e é de 8 bits, RISC, com 32KB de memória flash, 1 KB de EEPROM, 2KB de SRAM, 32 registradores de uso geral, 3 temporizadores/contadores, uma Usart, portas para comunicação SPI, 6 conversores AD de 10bits e um watchdog timer, entre outras características. A tensão de operação dele é entre 1,8 e 5,5 V.

O *software* que controla os pinos e ações do Arduino também podem ser desenvolvidos no programa chamado Arduino IDE, e aceita códigos em C++.

Conforme explanação de Hoepers (2012), o Arduino possui uma comunidade ativa que trabalha em sua evolução agregando novas funcionalidades ao projeto. Comumente são encontrados dispositivos que se encaixam na placa Arduino de forma a permitir novas possibilidades e facilitar a vida dos projetistas em seus protótipos. Esses dispositivos são chamados de Shields, abrem novos horizontes aos desenvolvedores agregando novas funcionalidades ao Arduino.

Abaixo alguns dos inúmeros *Shields*, que são placas de hardware que podem ser plugadas no Arduino, criados para usar em conjunto com o Arduino são: (SPARKFUN, 2011).

- Ethernet: Permite que um Arduino seja conectado a internet utilizando uma biblioteca de rede que suporta tanto TCP

(Transmission Control Protocol) como UDP (User Datagram Protocol);

- Kit Joystick: Permite adotar um Arduino com um joystick se encaixando sobre o mesmo transformando em um controle simples, contando 4 botões e 1 de seleção do joystick, as possibilidades são inúmeras com este shield;
- MicroSD: Este permite aumentar a capacidade de memória usada pelo projeto ou pode funcionar como registrador de logs onde se pretenda gravar determinadas informações, como por exemplo a leitura de um sensor;
- Ardumoto: Este permite ao Arduino controlar dois motores de corrente contínua (DC), agilizando projetos de robótica que envolve este tipo de controle;
- Wifly: Este shield permite ao Arduino conectar a redes sem fio (Wireless) usando o protocolo 802.11 b/g, auxiliando os projetos que envolvem conexão sem fio.

O arduino por ser implementado em C++ , possui princípios de programação orientada a objetos, especialmente em suas bibliotecas. Vejamos agora os principais fundamentos da P.O.O. (programação orientada a objetos) e como isso se aplica no Arduino.

Conforme dito por Oliveira e Carvalho (2014), as unidades fundamentais da POO são as classes e os objetos. O conceito de classe é uma abstração que lida com funcionalidades determinadas, e suas instâncias, que geram objetos. Classes nos permitem criar abstrações, com atributos e métodos, de forma a termos um modelo que possui estruturas em comum e é capaz de gerar versões diferentes com essas mesmas estruturas, que são os objetos. Usando uma analogia, poderíamos ter a classe “pessoa”, com os seus atributos básicos, dois olhos, cor de cabelo e etc, além de suas ações ou métodos, que são as suas interações, como andar, dirigir, e etc. Cada pessoa possui os mesmos atributos, porém a cor dos cabelos, o tamanho dos olhos e a forma como cada um dirige, são informações particulares de cada pessoa. Assim, apesar de terem uma mesma estrutura, as pessoas podem ter versões diferentes. Assim são as classes e os objetos, sendo as classes as abstrações que

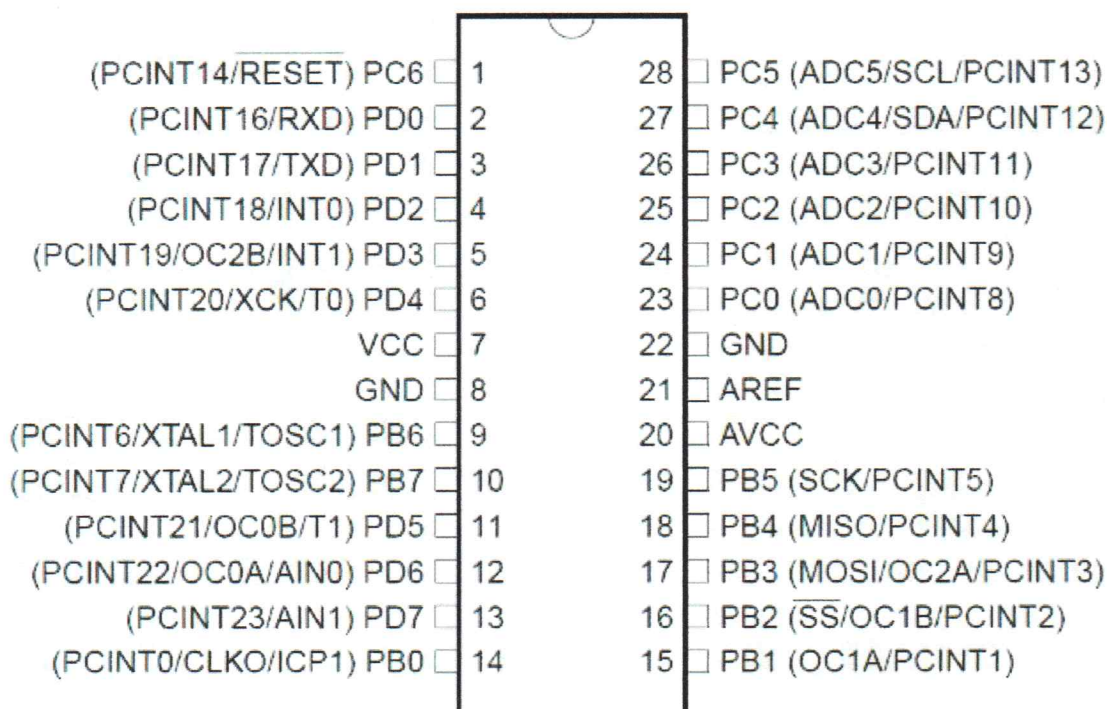
compartilham de estruturas em comum e os objetos as derivações dessas estruturas, podendo receber diversos valores para os seus atributos e métodos.

Oliveira e Carvalho (2014) acrescentam ainda sobre a utilização de classes que é também um forte aliado para a modularização do código. Classes lidam geralmente com domínios específicos, de forma que as boas práticas do código limpo nos fazem designar uma classe para cada função/ação específica. Nas classes, os seus comportamentos e variáveis são inicializados por estruturas conhecidas como construtores, que são as primeiras funções a serem executadas em uma classe, realizando as declarações iniciais de seus atributos.

E ainda sobre C++, Oliveira e Carvalho (2014) dizem que existem dois tipos de arquivos principais, quando lidamos com classes, “.cpp” e o “.h”. Os arquivos “.h”, são arquivos de definição, onde declaramos as nossas classes. Implementamos as nossas classes nos arquivos “.cpp”. Todas as classes possuem dois campos, privados e públicos. Campos públicos são informações que outras classes e trechos do código podem utilizar. Campos privados são informações relativas apenas a aquela classe.

Figura 13 - Chip ATmega328

Pinout ATMEGA328P



Fonte: http://1.bp.blogspot.com/-brED5sue_wU/VR1pCVjt5fI/AAAAAAAAACV8/ins7r1JQ_yU/s1600/328p-pinout.png

(2016)

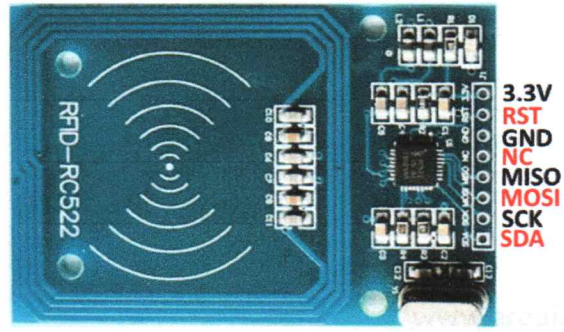
Carvalho e Oliveira (2014) mostram que os registradores de PORT possibilitam a manipulação mais ágil dos pinos de I/O do microcontrolador em uma placa Arduino. Que, sendo assim, ao invés de usar `DigitalWrite()` e `DigitalRead()`, é vantajoso usar diretamente os registradores `DDRx`, `PORTx` e `PINx`. Os chips usados na placa Arduino, que são o ATmega8 e a ATmega168, possuem três PORTS, que são a PORT B, no pino digital de 8 a 13; a PORT C, que são os pinos de entrada analógica; e, a PORT D, que pinos digitais 0-7. Então, cada uma é controlada por três resistradores diferentes: a DDR, o PORT e o PIN, que são definidos na linguagem arduino. O registrador DDR ira determinar se o pino é uma entrada ou saída. O registrador PORT vai controlar o nível em que o pino está, sendo o nível alto (1) ou o nível baixo (0), assim como o DDR pode ser tanto de escrita como de leitura. E por fim, o registrador PIN, que irá ler o estado dos pinos deliberados como entrada através da função `pinMode()`, sendo então, apenas para leitura.

Citado por Carvalho e Oliveira (2014) ainda, diz que a PORTD está mapeado no arduino para os pinos digitais 0-7 na placa. A PORTB mapeia os pinos 8 até 13. Os dois bits mais altos (6 e 7) mapeiam os pinos do cristal e não são utilizáveis, visto que já tem um cristal de quartzo de 16MHz ligados a eles. A PORTC mapeia os pinos analógicos do 0 até 5. Cada bit destes registradores correspondem a um único pino do chip.

O DDRD é o registrador de controle de direção para PortD, onde os bits definem as entradas ou saídas.

Como mostrado no site Arduin e Cia (2014), o módulo RFID RC522 utiliza a interface SPI para comunicação com o Arduino por meio dos pinos digitais de 10 à 13, além do pino 9 ligado ao RST do módulo. Veja na imagem abaixo a disposição dos pinos e a ligação ao Arduino, com os pinos suportados pela biblioteca MFRC522, que é utilizado no programa de controle de acesso. A alimentação do módulo, que é feita por meio do pino 3.3V do Arduino.

Figura 14 - Pinagem do modulo RFID



Módulo RFID RC522	Arduino
3.3	Pino 3.3V
RST	Pino 9
GND	Pino GND
NC	Não conectado
MISO	Pino 12
MOSI	Pino 11
SCK	Pino 13
SDA	Pino 10

Fonte: <http://1.bp.blogspot.com/-grDlvrxyaE/VInGrHNX85I/AAAAAAAAADM/GucLLVdkah4/s1600/Tabela-pinos-RFID-RC522.PNG>
(2016)

4.4. Banco de Dados

O sistema de banco de dados funciona basicamente como um sistema de registro de dados. Conforme dito por Heuser (2008) banco de dados é o conjunto de arquivos integrados que atendem a um conjunto de sistemas. Através disso, surgem então os Sistemas de Gerência de Banco de Dados (SGBD), que é oriundo da necessidade de melhorar o controle de dados e diminuir problemas com a complexidade dos bancos de dados, sabendo que o compartilhamento de dados influencia na estrutura dos *softwares*.

Segundo Heuser (2008), os SGBD funcionam como um *software* que incorpora as funções de definição, recuperação e alteração de dados em um banco de dados.

4.4.1. Sistema de Gerenciamento de Banco de Dados – SGBD

A utilização de banco de dados está inserida na realidade e no cotidiano da sociedade. Muitas vezes sendo utilizadas de maneira simples, que pode ser no armazenamento de de telefone ou *email*, ou de maneira mais complexa, como em operações nos caixas eletrônicos.

Um sistema de gerenciamento de banco de dados (SGBD) pode ser definido como uma coleção de programas que permite aos usuários criar e manter um banco de dados. O SGBD é, portanto, um sistema de software de propósito geral que facilita os processos de definição, construção, manipulação e compartilhamento de bancos de dados entre vários usuários e aplicações. A definição de um banco de dados implica especificar os tipos de dados, as estruturas e as restrições para os dados a serem armazenados em um banco de dados. (ELMASRI, 2005)

Existe no mercado atual diversos SGDB's, visto que vem trazendo praticidade e funcionalidades. Um exemplo é o MySQL, utilizado neste trabalho.

4.4.2. MySQL

O MySQL foi criado pela empresa MySQL AB, que era uma empresa instituída na Suécia e tinha como principais fundadores, David Axmark, Allan Larsson e Michael “Monty” Widenius.

“Quando começamos, tínhamos a intenção de usar o mSQL para conectar às nossas tabelas utilizando nossas rápidas rotinas de baixo nível (ISAM). Entretanto, depois de alguns testes, chegamos a conclusão que o mSQL não era rápido e nem flexível o suficiente para nossas necessidades. Isto resultou em uma nova interface SQL para nosso banco de dados, mas com praticamente a mesma Interface API do mSQL. Esta API foi escolhida para facilitar a portabilidade para códigos de terceiros que era escrito para uso com mSQL para ser portado facilmente para uso com o MySQL”. (AXMARK; LARSSON; WIDENIUS; 2010)

Conforme dito por Neto (2011), foi em Janeiro de 2008, que a MySQL AB, foi adquirida pela Sun Microsystems, por US\$ 1 bilhão, um preço exagerado para um setor que trabalhava com licenças livres. E então no dia 20 de Abril de 2009 a Oracle adquiriu a Sun Microsystems, assim como todos os seus produtos, inclusive o MySQL. Porém, ainda assim, o MySQL continuava possuindo a licença GNU, que é uma denominação de licença para todo software livre, baseando-se em alguns critérios, tais como, a liberdade de executar o programa, para qualquer propósito (liberdade nº 0), a de estudar como o programa funciona, assim adaptando-o de acordo com as suas necessidades (liberdade nº 1), sendo o acesso ao código-fonte um pré-requisito para esta liberdade. Tendo ainda como critério, a liberdade de redistribuir cópias a fim de ajudar ao próximo (liberdade nº 2), e por fim, a liberdade de aperfeiçoar o programa e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie deles (liberdade nº 3). O acesso ao código-fonte é um pré-requisito para esta liberdade.

Alecrim (2008), diz que “O MySQL é um dos sistemas de gerenciamento de banco de dados mais populares que existe e, por ser otimizado para aplicações Web, é amplamente utilizado na Internet”. É desenvolvido utilizando o princípio do código livre, além de possuir também uma licença comercial, que quer dizer, ter uma versão

paga, a fim de garantir suporte diferenciado dos desenvolvedores. O que diferencia de muitos outros aplicativos *open source*.

Como visto no site da DEVMEDIA (2016), o MySQL tem quase tudo que os concorrentes têm, com alguns vantagens ainda, como por exemplo, ser gratuito (não totalmente), possuir código fonte aberto, ser leve, rápido, seguro e prático. Tendo como principais vantagens o seu desempenho, pois é considerado o banco de dados mais rápido, sendo a agilidade um ponto relevante quando se fala em sistemas, principalmente na *web*, a performance também vem se mostrando muito calcada; outra vantagem é a segurança, por possuir diversos tipos de tabelas, que é uma característica exclusiva do MySQL, é possível ter um BD muito seguro e estável contando com integridade referencial, *backup* e *restore*, controle de usuário e acessos, sendo possível ainda, caso haja necessidade, verificar e realizar correções de corrompimento de tabelas; a sua aplicabilidade, pois pode ser usado como BD *desktop*, para aplicações corporativas também, dando suporte de diversas linguagens de programação; por fim, ressaltando a vantagem de ser gratuito, pois está inscrito no tipo de licenciamento GNU General Public License.

4.4.3. UML

Conforme Guedes (2011) a UML (*Unified Modeling Language* ou Linguagem Unificada de Modelagem) se tornou nos últimos anos uma linguagem padrão para modelar softwares, essa é uma linguagem visual baseada nos paradigmas de orientação a objeto.

Segundo Wazlawick (2011) a UML foi aos poucos sendo definida em 1994, com a junção das linguagens de diagramas de James Rumbaugh e Grady Biich, e a partir dessa ligação foi criada a empresa Rational, que depois disso, Ivar Jacobson se uniu a Raumbauhg e Biich na empresa e agregou seus casos de uso e outras notações a linguagem que vinha sendo desenvolvida.

Wazlawick (2011) ainda explica que os diagramas UML são divididos em três famílias de diagramas, que são:

- **Diagramas estruturais**, definidos com os diagramas de pacotes, objetos, estrutura composta, componentes e distribuição;

- **Diagramas Comportamentais**, que são os diagramas de casos de uso, atividade e máquina de estados;
- **Diagramas de Interação**, que compreende os diagramas de comunicação, sequência, tempo e visão geral de integração.

Ainda segundo Wazlawick (2011) não é necessário a utilização de todos os diagramas durante a etapa de desenvolvimento de um sistema. Nesse trabalho foi utilizado apenas um diagrama.

4.4.3.1. Diagrama de Classe

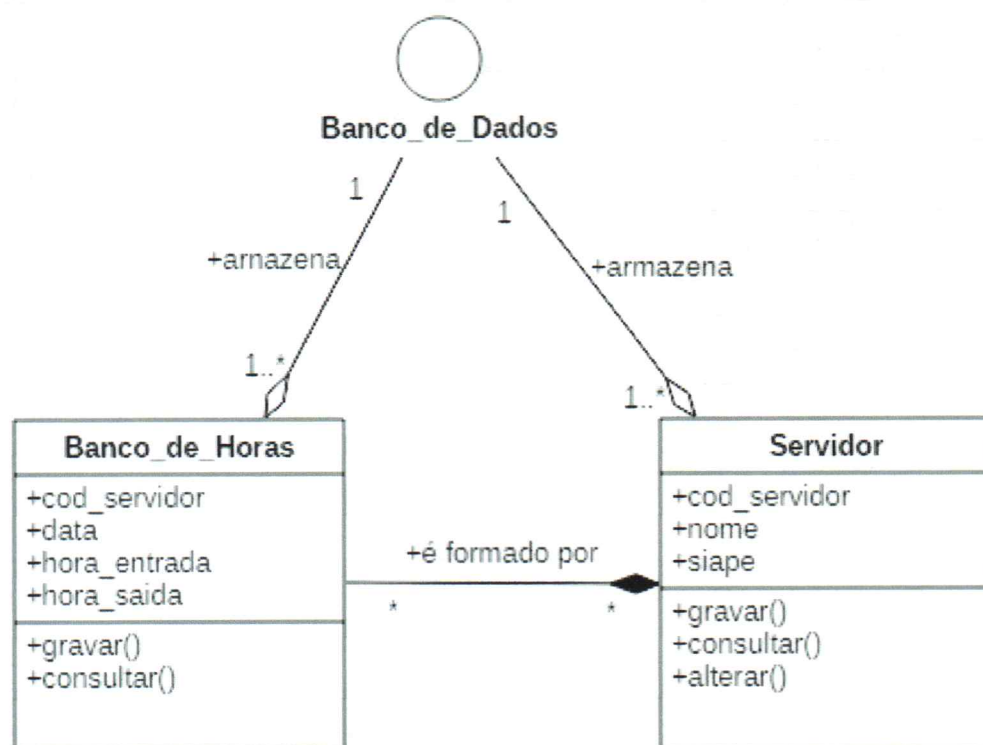
Os diagramas de classes para Guedes (2012) é certamente o mais utilizado e visto como um dos mais importantes dessa linguagem de modelagem, servindo ainda de apoio para a maioria dos outros diagramas.

O diagrama define a estrutura das classes utilizadas pelo sistema, determinando seus atributos e seus métodos, além de estabelecer as relações entre as outras classes (GUEDES, 2012).

“Os diagramas de classes são utilizados no desenvolvimento de um modelo de sistema orientado a objetos para mostrar as classes em um sistema e as associações entre essas classes.” (SOMMERVILLE, 2011, p. 90).

Na figura 15, mostrada a seguir, é o diagrama de classes. Onde é possível visualizar as classes e observar que essas classes possuem atributos e métodos e associações entre elas.

Figura 15- Diagrama de Classes



Fonte: O autor

4.5. Uso da tecnologia da gestão empresarial

Um dos grandes vetores das transformações no cenário competitivo é a contínua evolução da tecnologia que, em virtude de sua grande disseminação, afetou de modo significativo todas as atividades humanas, e fez crescer o grau de incerteza e imprevisibilidade do futuro. Dentre as novas tecnologias, destaca-se a Tecnologia da Informação (TI), que passou a ser um importante fator competitivo para as organizações (ALBANO, 2001).

O uso bem planejado de Tecnologia da Informação dará suporte aos três principais papéis que exerce o Sistema de Informação na organização, que são: busca de vantagem competitiva, apoio à tomada de decisão gerencial e apoio às operações. Conforme mencionado por LAUDON & LAUDON (2004) a tecnologia da Informação é uma das muitas ferramentas que os gerentes utilizam para enfrentar as mudanças.

Utilizar as tecnologias da informação e integrar entre a sua estratégia e a do negócio ultrapassa a ideia de produtividade, tornando-se um fator importante para o sucesso da empresa. Não somente o *hardware* e *software* usados, mas sim o alinhamento da TI com as estratégias da empresa e sua estrutura organizacional. Conforme foi dito por ROCKART et al. (1996), "In sum, the load on IT organizations is heavier than ever before, and the management of IT is more complex".

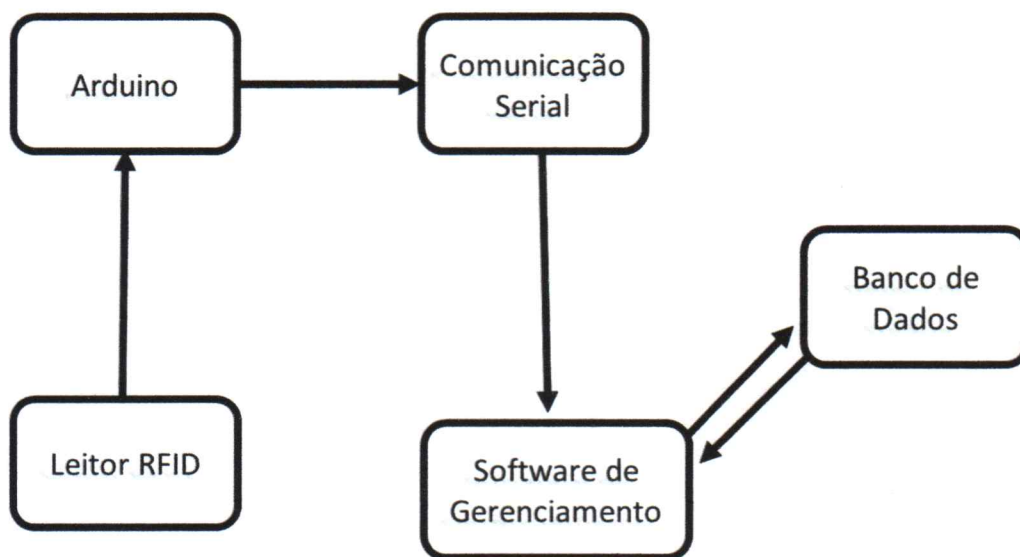
O ponto eletrônico surge como uma ferramenta que traz benefícios para a gestão empresarial. Visto que é uma ferramenta para melhorar o controle, monitoramento e assegurar melhor o empregador e o empregado.

5. METODOLOGIA

Através de pesquisa bibliográfica sobre o surgimento do Ponto Eletrônico, bem como as plataformas existentes para desenvolvimento do protótipo, assim abordando sobre o Arduino, utilizando a versão 1.6.0, visto que é uma ferramenta mais barata e que utiliza uma linguagem de códigos aberta e pode ser usada em qualquer SO. Explicações sobre o Banco de Dados para armazenar as informações recebidas. De acordo com Echer (2001), para o pesquisador redigir o projeto de pesquisa é necessário um embasamento teórico sobre o tema e a metodologia a ser utilizada, conhecimento de como o projeto deve ser estruturado, e isto se consegue com a revisão de literatura. Seguindo então para a construção do protótipo em si, que foi a parte prática do trabalho e resultou na construção do protótipo.

Funcionara conforme mostrado no diagrama da figura 16 a seguir:

Figura 16 - Diagrama do funcionamento do projeto



Fonte: O autor

O sistema foi construído através da montagem dos equipamentos da placa do arduino, com o leitor RFID, utilizando jumper macho-femea, usando para intermédio uma placa *protoboard*. Então foi montado o sistema de banco de dados, utilizando o

wamp, através do PhpAdmin, juntamente a um código inserido no *software* Arduino e no netbeans. Com a ligação desses processos, foi possível construir o protótipo.

O arduino envia para o software de gerenciamento os dados da TAG através da comunicação serial. E o software de gerenciamento armazena as informações no BD. Esse leitor irá receber a informação do cartão, que através da comunicação serial também, irá transmitir a informação para a placa arduino, seguindo para o bando de dados. Que irá armazenar a informação coletada, possibilitando acesso a essa informação no software de gerenciamento do banco de dados.

Por meio da ferramenta JDBC (Java Database Connectivity) que é um conjunto de classes e interfaces escritas em Java que fazem o envio de instruções SQL (Structure Query Language) para qualquer banco de dados relacional. Essas ferramentas foram selecionadas devido a sua acessibilidade e usabilidade. Utilizando também o WampServer (Windows, Apache, MySQL, PHP), sendo um pacote de programas que instala automaticamente o Apache, PHP, MySQL database, PHPmyadmin e SQLitemanager. Possibilitando criar um servidor para armazenar os dados em um banco de dados. Utilizando também o StarUML para criação do diagrama de classes.

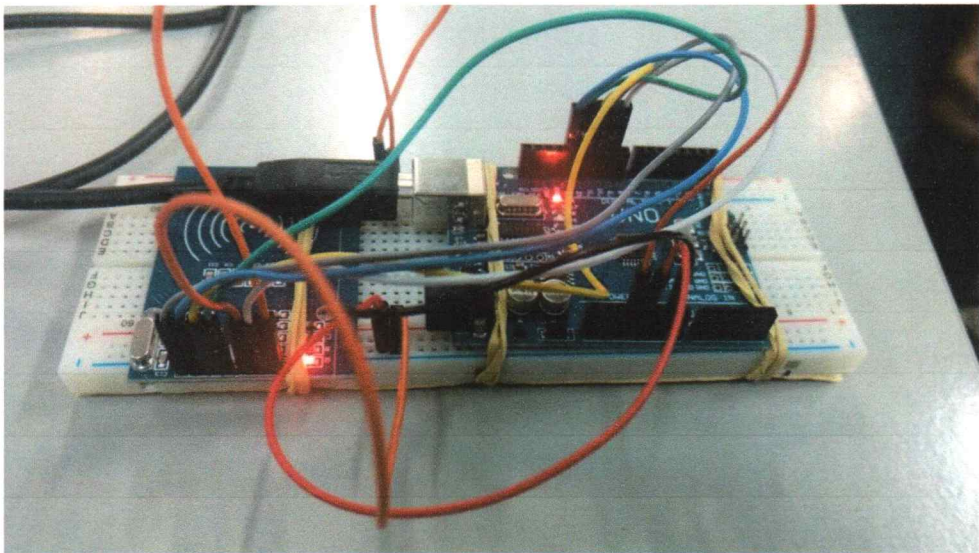
Utilizando o NetBeans IDE 8.1, desenvolvendo a programação em JAVA, para a comunicação serial. Como também, fazer o download da biblioteca MFRC522, descompactando o arquivo, para renomear a pasta rfid-master para MFRC522, e colocar dentro da pagina LIBRARIES da IDE do arduino.

Com a utilização dessas ferramentas foi possível criar o bando de dados onde consegue-se armazenar, alterar e cadastrar informações oriundas do protótipo do Ponto Eletrônico.

6. RESULTADOS OBTIDOS

Quanto a montagem do hardware foram feitas ligações entre placa do arduino, o protoboard e o leitor do RFID, como mostrado na figura 17, mostrada a seguir:

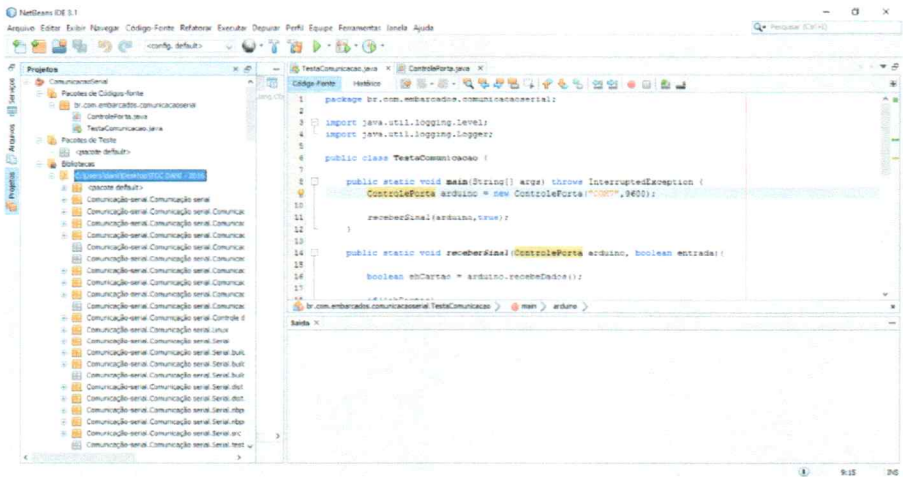
Figura 17 - Prototipo: Ponto Eletrônico Arduino



Fonte: O autor

Conforme mostrado na figura 18, utilizando a ferramenta NetBeans,afim de fazer comunicação serial.

Figura 18 - Tela do NetBeans: Comunicação Serial



Fonte: O autor

Dessa maneira, é identificado a informação recebida pela leitura do cartão no leitor RFID, definindo se o valor recebido será *true*, aceitando o acesso, informando que o acesso foi liberado, ou *false*, que não reconhece a leitura e informa que o acesso foi negado.

Utilizando esses códigos juntamente a montagem do hardware, será possível realizar a leitura do cartão (figura 19) no leitor RFID.

Figura 19 - Personalização do Cartão Eletrônico



The image shows a personalized electronic card for Daniela Ferraz. The card has a green background. At the top left, there is a logo consisting of a grid of green squares with a red square in the top-left corner. To the right of the logo, the text reads: "INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SERTÃO PERNAMBUCANO CAMPUS FLORESTA". Below the logo is a portrait of Daniela Ferraz, a woman with dark hair and a smile. Below the portrait, the name "Daniela Ferraz" is written in a large, bold, black font, and "Gestora de TI" is written in a smaller, bold, black font. To the right of the card, there is a form with five input fields, each with a label and a value. The labels are: "Nome Completo:", "CPF:", "Função:", "SIAPE:", and "Tipo Sanguíneo:". The values are: "Daniela Nogueira Ferraz Cornélio", a blurred blue box, "Gestora de TI", a blurred blue box, and "+ A". At the bottom of the form, there is a red RFID symbol consisting of three curved lines on each side of the word "RFID" in bold black font.

Nome Completo:	Daniela Nogueira Ferraz Cornélio
CPF:	[Blurred]
Função:	Gestora de TI
SIAPE:	[Blurred]
Tipo Sanguíneo:	+ A

RFID

Fonte: O autor

Para armazenamento dessas informações, foi utilizado o banco de dados Wamp.

7. CONSIDERAÇÕES FINAIS

7.1. Considerações gerais

A primeira consideração é que o protótipo funcionou como projetado.

Sabendo-se que o ponto eletrônico é de suma importância para o melhor controle de acesso e auxílio do gerenciamento de pessoas nas organizações, além de ser mais prático e seguro, armazenando e registrando informações da rotina diária, possibilitando consultas futuras.

7.2. Considerações sobre o *hardware*

A dificuldade de encontrar os materiais necessários para a construção do protótipo acarretou em um certo atraso na montagem. Porém, foi possível comprar todas as ferramentas necessárias.

Apesar de não ter um visual tão atrativo, o *hardware* está funcionando como pretendido.

7.3. Considerações sobre o *software*

Foram utilizado *softwares* livres, facilitando o *download*. Havendo tempo habio para elaboração dos códigos. Ficando apenas como atraso, os testes com o *hardware* arduino, pela dificuldade de encontrar os materiais.

7.4. Trabalhos futuros

Esse trabalho abre inúmeras opções de futuros trabalhos seguindo o mesmo racionio. Poderia ser acrescentado um atuador a essa aplicação, como também, construído um *software* em que os servidores poderiam consultar seu banco de horas, assim como o administrador pudesse também consultar, imprimir e manusear melhor as informações armazenadas.

8. REFERÊNCIAS

ABREP. **Pesquisa de satisfação realizada com usuários do SREP**. Disponível em: <<http://www.abrep.com.br/site/pesquisa-de-satisfacao-realizada-com-usuarios-do-srep/>> Acesso em: 09 de agosto de 2016.

ALBANO, Cláudio S. **Problemas e ações na adoção de novas tecnologias de informação: um estudo em cooperativas agropecuárias do Rio Grande do Sul**. Dissertação de Mestrado PPGA/EA/UFRGS. Porto Alegre: UFRGS, 2001, 122 p. Acesso em 04 ago 2016

ALECRIM, Emerson. **Banco de dados MySQL e PostgreSQL**. [S.l.], 2008. Disponível em: < <http://www.infowester.com/postgremysql.php>>. Acesso em: 21 ago. 2016.

ANTUNES, H; WERNER, H.; ALVES, L. 2011. **O sistema de RFID Digital**. 2011. Disponível em < http://www.gta.ufrj.br/grad/11_1/rfid/sistema_rfid_digital.html> acesso em 15 set 2016.

ARDUINO – **Introduction**. **Arduino**. [S.l.]. Disponível em: <https://www.arduino.cc/en/Guide/Introduction>. Acesso em: 02 set. 2016.

ARDUINO e Cia. **Controle de acesso com módulo RFID RC522**. 2014. Disponível em:<<http://www.arduinoecia.com.br/2014/12/control-de-acesso-modulo-rfid-rc522.html>> Acesso em: 24 set 2016.

AXMARK, D.; LARSSON, A.; WIDENIUS, M. **Site de documentação sobre o MySQL**. Disponível em: <http://dev.mysql.com/>, Acesso em 10 set 2016.

Barge P, Gay P, Merlino V, Tortia C (2013) **Radio frequency identification technologies for livestock management and meat supply chain traceability**. Canadian Journal of Animal Science. 93:23-33.

BERNARDO, Cláudio Gonçalves. **A Tecnologia RFID e os benefícios da etiqueta inteligente para os negócios**. Revista Eletrônica UNIBERO de Produção Científica. São Paulo, 2004. Disponível em <<http://aulas->

gelsimar.googlecode.com/svn/trunk/TOLER%C3%82NCIA_FALHAS/A_Tecnologia_RFID.pdf> Acesso em 10 de maio de 2016.

BRASIL. Ministério do Trabalho e Emprego. **PORTARIA Nº 1.510, DE 21 DE AGOSTO DE 2009** Brasília: Ministério do Trabalho e Emprego, 2009. Disponível em: <http://www.trtsp.jus.br/geral/tribunal2/ORGaos/MTE/Portaria/P1510_09.html>. Acesso em: 17 set. 2016.

CARVALHO, Fabrício F. e OLIVEIRA, Andre A. **TuKolk: Um jogo de console utilizando a plataforma Arduino**. Rio de Janeiro. 2014. Disponível em: <<http://bsi.uniriotec.br/tcc/201408OliveiraKolk.pdf>> Acesso em: 12 set 2016.

CNRFID - Centre national de référence RFID. **Classification des tags RFID**. 2016. Disponível em: <<http://www.centrenational-rfid.com/classification-des-tags-rfid-article-19-fr-ruid-17.html>> Acesso em: 12 set 2016.

DALFOVO, O.; HOSTINS, C. A. 2010. **Delineamento para aplicação do RFID na logística de supermercado como inteligência competitiva**. Blumenau, SC. 2010. Disponível em: <<http://rica.unibes.com.br/index.php/rica/article/viewFile/379/333>> acesso em 12 ago 2016.

DALFOVO, O.; TAMBORLIN, N. **Business Intelligence** 1º Edição. Blumenau. 2010.

DEVEMEDIA. **MySQL: Quem é você?**. 2016. Disponível em: <<http://www.devmedia.com.br/mysql-quem-e-voce/1752>> Acesso em: 21 ago 2016.

DIAS, R. R. F. **RFID Implementation**. São Paulo. 2008.

ECHER, I. C. 2001. **A revisão de Literatura na Construção do Trabalho Científico**. Porto Alegre, RS. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/23470/000326312.pdf?sequen>> acesso em: 06 ago 2016.

ELMASRI, Ramez, **Sistemas de banco de dados**. 4. ed. São Paulo: Pearson Addison Wesley, 2005.

FIGUEIREDO, T. B. **Aplicações de tecnologias sem fio em operações logísticas. Dissertação de Mestrado.** Departamento de Engenharia Industrial da Pontifícia Universidade Católica do Rio de Janeiro. Rio de Janeiro, 2004.

FINKENZELLER, K. 2010 apud MARQUES, J. M. C. 2012. **Análise dos fatores que influenciam a eficiência da tecnologia RFID aplicada a sistemas de produção.** Porto Alegre, RS. 2012 p. 28. Disponível em: <<https://www.lume.ufrgs.br/bitstream/handle/10183/75902/000884115.pdf?sequence=1>> acesso em: 12 ago 2016.

GUEDES, Gileanes T. A. **UML Uma abordagem Prática 2.** 2ª edição. Novatec, 2011.

GLOVER, B.; BHATT, H. **Fundamentos de RFID.** Rio de Janeiro: Alta Books, 2007. 228 pp.

HEUSER, C. A. **Projeto de banco de dados.** 6. ed. Porto Alegre: Bookman, 2008.

HOEPERS, Rodrigo. **Veículo autônomo usando arduino.** Relatório. Disponível em: <<http://siaibib01.univali.br/pdf/Rodrigo%20Hoepers.pdf>> Acesso em: 20 ago 2016.

INTERMEC.com. 2016. **Fundamentos da RFID: Entendendo e usando a identificação por radiofrequência.** Disponível em: <http://www.intermec.com/public-files/white-papers/br/wpABC_BR.pdf> Acesso em: 17 set 2016.

LAUDON, K. C., LAUDON, J. P., **Sistemas de Informação Gerenciais: Administrando a empresa digital.** 5. ed. São Paulo. Prentice-Hall, 2004.

MAUSHAMMER, John. **Sparkfun Autonomous Vehicle Competition.** 2009. Disponível em: <<http://makezine.com/2009/04/13/sparkfun-autonomous-vehicle-competi/>> Acesso em: 12 set 2016.

Microchip Atmel. **ATmega328.** Ficha de dados. Disponível em: <<http://www.atmel.com/pt/br/devices/ATMEGA328.aspx>> Acesso em: 15 set 2016.

NAZARENO, Gustavo S. **Implantação de uma Central de Service Desk à luz do ITIL v3: propostas para as gerências de portfólio, demanda, incidentes e problemas.** 09 de Agosto de 2013. 48f. Dissertação - Serviço Nacional de

Aprendizagem Comercial – SENAC, Unidade EAD – SENAC/DF. Disponível em: <
<http://www.edilms.eti.br/uploads/file/orientacoes/GTIDF05%20-%20Gustavo%20Setubal%20Nazareno.pdf>> Acesso em: 02 de maio 2016.

Neto, S V. B. MYSQL x FIREBIRD - **Uma Análise Comparativa de Desempenho**. Guaratinguetá, 2011. 66p. Monografia, Faculdade de Tecnologia de Guaratinguetá.

Oliveira, André Alves, Fabrício Kolk Carvalho, and Leila Cristina Vasconcelos de Andrade. 2014. Disponível em: <<http://bsi.uniriotec.br/tcc/201408OliveiraKolk.pdf>> Acesso em: 06 set 2016

Pimenta, Rodrigo. **Relógio de ponto: o encontro da tradição com a inovação**. Artigo. 2011. Disponível em: <<http://www.administradores.com.br/mobile/artigos/tecnologia/relogio-de-ponto-o-encontro-da-tradicao-com-a-inovacao/59467/>> Acesso em: 10 set 2016.

PORTARIA MINISTRO DE ESTADO DO TRABALHO E EMPREGO Nº 1.510 DE 21.08.2009 . D.O.U.: 25.08.2009. Disponível em: <
http://www.normaslegais.com.br/legislacao/portaria1510_2009.htm > Acesso em: 05 set 2016.

PRADO, N.R.S.A; PEREIRA, N.A. & POLITANO, P.R. **Dificuldades para a adoção de RFID nas operações de uma cadeia de suprimentos**. Anais. XXVI ENEGEP – Encontro Nacional de Engenharia de Produção, Fortaleza, 2006.

PRERADOVIC, S.; KARMARKAR, N. 2006 apud MARQUES, J. M. C. 2012. **Análise dos fatores que influenciam a eficiência da tecnologia RFID aplicada a sistemas de produção**. Porto Alegre, RS. 2012 p. 29. Disponível em: <
<https://www.lume.ufrgs.br/bitstream/handle/10183/75902/000884115.pdf?sequence=1>> acesso em: 16 ago 2016.

RFID JOURNAL BRASIL, 2011. Disponível em <
http://brasil.rfidjournal.com/perguntas_frequentes > Acesso em: 11 de maio de 2016.

RFID JOURNAL. **HP implanta tecnologia RFID em toda a cadeia de suprimentos.** Disponível em: < <http://www.rfidjournal.com/article/view/8448/1>>. Acesso em: 20 ago 2016

RFIDJOURNAL.com, 2008 apud DIAS. R. R. F. **RFID Implementation.** São Paulo. 2008 p. 3

ROBERTS, C, 2006 apud MARQUES, J. M. C. 2012. **Análise dos fatores que influenciam a eficiência da tecnologia RFID aplicada a sistemas de produção.** Porto Alegre, RS. 2012 p. 29. Disponível em: < <https://www.lume.ufrgs.br/bitstream/handle/10183/75902/000884115.pdf?sequence=1>> acesso em: 12 ago 2016.

ROCKART, J.F.; EARL, M.J. & ROSS, J.W.: **"Eight Imperatives for the new IT Organization"**. *Sloan Management Review*, v.38, n.1, p.43-55, Fall 1996.

ROSA. L. A. **Aplicação do RFID na cadeia logística.** Editora EPUSP. São Paulo. 2006.

SOUZA, D. A. M.; SOUZA, H. R.; SÁ, M. A. L.; CINTRA, S. P. V. **A Logística na Gestão de Estoque por Meio da Identificação por Rádio Frequência (RFID).** SEGET – Simpósio de Excelência em Gestão e Tecnologia, 2009. Disponível em: <http://www.aedb.br/seget/artigos09/233_233_A_Logistica.pdf>. Acesso em: 08 set 2016.

STANTON, M. - **A identificação por radio frequência está chegando.** Jornal O Estado de São Paulo, Coluna Tecnologia, 2004. Disponível em <www.estadao.com.br>. Acesso em: 21 ago 2016.

THORNTON, F. 2006 apud MARQUES, J. M. C. **Análise dos fatores que influenciam a eficiência da tecnologia RFID aplicada a sistemas de produção.** Porto Alegre, RS. 2012 p. 27.

Wazlawick, R. S. **Análise e projeto de sistemas de informação orientados objetos.** 2ª edição. Rio de Janeiro. Elsevier. 2011.

APÊNDICES

APENDICE A – Código para controle de porta:

```
package br.com.embarcados.comunicacaoserial;

import gnu.io.CommPortIdentifier;

import gnu.io.NoSuchPortException;

import gnu.io.SerialPort;

import java.io.IOException;

import java.io.InputStream;

import java.nio.charset.Charset;

import java.util.logging.Level;

import java.util.logging.Logger;

//import java.io.OutputStream;

import javax.swing.JOptionPane;

import sun.misc.IOUtils;

public class ControlePorta {

    //private OutputStream serialOut;

    private int taxa;

    private String portaCOM;

    private InputStream serialIn;
```



```
/**  
  
 * Construtor da classe ControlePorta  
  
 * @param portaCOM - Porta COM que será utilizada para enviar os dados para o arduino  
  
 * @param taxa - Taxa de transferência da porta serial geralmente é 9600  
  
 */  
  
public ControlePorta(String portaCOM, int taxa) {  
  
    this.portaCOM = portaCOM;  
  
    this.taxa = taxa;  
  
    this.initialize();  
  
}  
  
/**  
  
 * Método que verifica se a comunicação com a porta serial está funcionando  
corretamente  
  
 */  
  
private void initialize() {  
  
    try {  
  
        //Define uma variável portId do tipo CommPortIdentifier para realizar a comunicação  
serial  
  
        CommPortIdentifier portId = null;  
  
        try {  
  
            //Tenta verificar se a porta COM informada existe  
  
            portId = CommPortIdentifier.getPortIdentifier(this.portaCOM);  
  
        }  
  
    }  
  
}
```

```

        catch (NoSuchPortException npe) {

            //Caso a porta COM não exista será exibido um erro

            JOptionPane.showMessageDialog(null, "Porta COM não encontrada.",

                "Porta COM", JOptionPane.PLAIN_MESSAGE);

        }

        //Abre a porta COM

        SerialPort port = (SerialPort) portId.open("Comunicação serial", this.taxa);

        serialIn = port.getInputStream();

        //serialOut = port.getOutputStream();

        port.setSerialPortParams(this.taxa, //taxa de transferência da porta serial

            SerialPort.DATABITS_8, //taxa de 10 bits 8 (envio)

            SerialPort.STOPBITS_1, //taxa de 10 bits 1 (recebimento)

            SerialPort.PARITY_NONE); //receber e enviar dados

    } catch (Exception e) {

        e.printStackTrace();

    }

}

/**
 * Método que fecha a comunicação com a porta serial
 */

public void close() {

```

```

try {

    //serialOut.close();

    serialIn.close();

} catch (IOException e) {

    JOptionPane.showMessageDialog(null, "Não foi possível fechar porta COM.",

        "Fechar porta COM", JOptionPane.PLAIN_MESSAGE);

}

}

/**

 * @param opcao - Valor a ser enviado pela porta serial

public void enviaDados(int opcao){

    try {

        serialOut.write(opcao);//escreve o valor na porta serial para ser enviado

    } catch (IOException ex) {

        JOptionPane.showMessageDialog(null, "Não foi possível enviar o dado. ",

            "Enviar dados", JOptionPane.PLAIN_MESSAGE);

    }

}

*/

public boolean recebeDados(){

    byte[] buffer = new byte[1024];

    int len = 0, aux = 0;

```



```
String id = new String();

try{

    while ( aux < 8 ){

        len = this.serialIn.read(buffer);

        id = id.concat(new String(buffer,0,len));

        aux += len;

    }

}catch ( IOException e ){

    e.printStackTrace();

}

return id.substring(0,1).equals("9") || id.substring(0,1).equals("y");

}

}
```

APÊNDICE B – Código para a comunicação serial

```
package br.com.embarcados.comunicacaoserial;

import java.sql.Date;
import java.sql.SQLException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.logging.Level;
import java.util.logging.Logger;

public class TestaComunicacao {

    static String dataPonto, horaEntrada, horaSaida;

    public static void main(String[] args) throws InterruptedException,
ClassNotFoundException, SQLException {
        ControlePorta arduino = new ControlePorta("COM7", 9600);

        receberSinal(arduino, true);
    }

    public static void receberSinal(ControlePorta arduino, boolean entrada) throws
ClassNotFoundException, ClassNotFoundException, SQLException, InterruptedException {

        boolean ehCartao = arduino.recebeDados();

        if (!ehCartao) {
            System.out.println("Acesso negado!");
        } else {
            System.out.println(entrada ? "Acesso liberado!" : "Até amanhã!");
            if (entrada) {
                String data = new Date(System.currentTimeMillis()).toString();
                dataPonto = data.replace("-", "/");

                SimpleDateFormat sdf = new SimpleDateFormat("HH:mm:ss");
                java.util.Date hora = Calendar.getInstance().getTime();
                horaEntrada = sdf.format(hora);
            }
        }
    }
}
```

```
} else {  
    SimpleDateFormat sdf = new SimpleDateFormat("HH:mm:ss");  
    java.util.Date hora = Calendar.getInstance().getTime();  
    horaSaida = sdf.format(hora);  
  
    ConexaoJDBC.baterPonto("1", dataPonto, horaEntrada, horaSaida);  
}  
  
// tenta parar a leitura por 10 segundos pra esperar proxima batida.  
if (entrada) {  
    entrada = false;  
    try {  
        Thread.sleep(10000);  
        receberSinal(arduino, entrada);  
  
    } catch (InterruptedException ex) {  
        Logger.getLogger(TestaComunicacao.class.getName()).log(Level.SEVERE,  
null, ex);  
    }  
}  
}  
  
System.exit(0);  
  
}  
}
```


APÊNDICE C – Conexão JDBC

```
package br.com.embarcados.comunicacaoserial;

import java.sql.Connection;

import java.sql.Date;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.text.SimpleDateFormat;

import java.util.Calendar;

/**
 *
 * @author danii
 */
public class ConexaoJDBC {

    private static final String USUARIO = "root";

    private static final String SENHA = "";

    private static final String URL =
"jdbc:mysql://localhost:3306/mysql?zeroDateTimeBehavior=convertToNull";

    private static final String DRIVER = "com.mysql.jdbc.Driver";

    public static void main(String[] args) throws ClassNotFoundException, SQLException {
```

```
listarServidores();
```

```
listarPontosBatidos();
```

```
//baterPonto();
```

```
}
```

```
public static Connection obterConexao() throws SQLException, ClassNotFoundException {
```

```
    Class.forName(DRIVER);
```

```
    return DriverManager.getConnection(URL, USUARIO, SENHA);
```

```
}
```

```
public static void listarServidores() throws ClassNotFoundException, SQLException {
```

```
    Connection dbConnection = null;
```

```
    Statement statement = null;
```

```
String selectTableSQL = "SELECT * FROM pontoeletronico.SERVIDOR";
```

```
try {
```

```
    dbConnection = obterConexao();
```

```
    statement = dbConnection.createStatement();
```

```
    // execute select SQL statement
```

```
    ResultSet rs = statement.executeQuery(selectTableSQL);
```

```
System.out.println("# RELAÇÃO DE SERVIDORES #");
```

```
    System.out.print("\n");
```

```
int qtd = 0;
```

```
while (rs.next()) {
```

```
    String userid = rs.getString("nome");
```

```

        String username = rs.getString("siape");

        String cod = rs.getString("cod");

        System.out.println("Nº: " + ++qtd);

        System.out.println("Nome: " + userid);

        System.out.println("Siape: " + username);

        System.out.println("Código: " + cod);

        System.out.print("\n");

    }

} catch (SQLException e) {

    System.out.println(e.getMessage());

} finally {

    dbConnection.close();

}

}

public static void listarPontosBatidos() throws ClassNotFoundException, SQLException {

    Connection dbConnection = null;

    Statement statement = null;

    String selectTableSQL = "SELECT * FROM pontoeletronico.bancodehoras";

    try {

        dbConnection = obterConexao();

        statement = dbConnection.createStatement();

        // execute select SQL statement

```



```
        ResultSet rs = statement.executeQuery(selectTableSQL);

        System.out.println("# RELAÇÃO DE PONTOS BATIDOS #");

        System.out.print("\n");

        int qtd =0;

        while (rs.next()) {

            String cod = rs.getString("cod_servidor");

            String entrada = rs.getString("hora_entrada");

            String data = rs.getString("data");

            String saida = rs.getString("hora_saida");

            System.out.println("Nº: "+ ++qtd);

            System.out.println("Código: " + cod);

            System.out.println("Data: " + data);

            System.out.println("Hora Entrada: " + entrada);

            System.out.println("Hora Saída: " + saida);

            System.out.print("\n");        }

    } catch (SQLException e) {

        System.out.println(e.getMessage());

    } finally {

        dbConnection.close();

    }

}
```

```
public static void baterPonto(String cod, String data, String entrada, String saida) throws  
ClassNotFoundException, SQLException {
```

```
    Connection dbConnection = null;
```

```
    try {
```

```
        dbConnection = obterConexao();
```

```
        PreparedStatement stmt = dbConnection.prepareStatement("insert into  
pontoeletronico.bancodehoras values (?, ?, ?, ?)");
```

```
        stmt.setString(1, cod);
```

```
        stmt.setString(2, data);
```

```
        stmt.setString(3, entrada);
```

```
        stmt.setString(4, saida);
```

```
        stmt.execute();
```

```
    } catch (SQLException e) {
```

```
        System.out.println(e.getMessage());
```

```
    } finally {
```

```
        dbConnection.close();
```

```
    }
```

```
}
```

```
}
```

APÊNDECE D – Código para o controle de Acesso – RFID – ARDUINO

```
#include <MFRC522.h>

//Programa : RFID - Controle de Acesso leitor RFID

//Autor : danielaFerraz

#include <SPI.h>

#include <MFRC522.h>

#define SS_PIN 10

#define RST_PIN 9

MFRC522 mfrc522(SS_PIN, RST_PIN); // Cria MFRC522 instância.

char st[20];

void setup()

{

  Serial.begin(9600); // Inicia a serial

  SPI.begin(); // Inicia SPI bus

  mfrc522.PCD_Init(); // Inicia MFRC522

  //Define o número de colunas e linhas do LCD:

}

void loop()

{

  // Procurar novos cartões

  if ( ! mfrc522.PICC_IsNewCardPresent())
```



```
{  
    return;  
}  
  
// Seleciona um dos cartões  
  
if ( ! mfrc522.PICC_ReadCardSerial())  
  
{  
    return;  
}  
  
//Mostra UID na serial  
  
String conteudo= "";  
  
for (byte i = 0; i < mfrc522.uid.size; i++)  
  
{  
    conteudo.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? "0" : ""));  
    conteudo.concat(String(mfrc522.uid.uidByte[i], HEX));  
}  
  
conteudo.toUpperCase();  
  
Serial.println(conteudo);  
}
```