

**UNIVERSIDADE FEDERAL DE SANTA MARIA  
COLÉGIO TÉCNICO INDUSTRIAL DE SANTA MARIA  
CURSO SUPERIOR DE TECNOLOGIA EM REDES DE  
COMPUTADORES**

**MIDDLEWARE DE RASTREABILIDADE PARA  
TECNOLOGIAS DE REDES SEM FIO**

**TRABALHO DE CONCLUSÃO DE CURSO**

**Rafael Copatti Trindade**

**Santa Maria, RS, Brasil  
2015**

**STRC/UFSM, RS TRINDADE, Rafael Copatti**

**Tecnólogo em Redes de Computadores**

**2015**

# **MIDDLEWARE DE RASTREABILIDADE PARA TECNOLOGIAS DE REDES SEM FIO**

**Rafael Copatti Trindade**

Trabalho de Conclusão de Curso (TCC) apresentado ao Curso Superior de Tecnologia em Redes de Computadores do Colégio Técnico Industrial de Santa Maria, da Universidade Federal de Santa Maria (UFSM,RS), como requisito parcial para obtenção de grau de

**Tecnólogo em Redes de Computadores**

**Orientador: Prof. Dr. Claiton Pereira Colvero**

**Co-Orientador: Prof. Msc. Miguel Bauermann Brasil**

**Santa Maria, RS, Brasil  
2015**

**UNIVERSIDADE FEDERAL DE SANTA MARIA  
COLÉGIO TÉCNICO INDUSTRIAL DE SANTA MARIA  
CURSO SUPERIOR DE TECNOLOGIA EM REDES DE  
COMPUTADORES**

A Comissão Examinadora, abaixo assinada,  
aprova o Trabalho de Conclusão de Curso

**MIDDLEWARE DE RASTREABILIDADE PARA TECNOLOGIAS DE  
REDES SEM FIO**

elaborado por  
**Rafael Copatti Trindade**

Como requisito parcial para a obtenção de grau de  
**Tecnólogo em Redes de Computadores**

**COMISSÃO EXAMINADORA:**

**Claiton Pereira Colvero, PhD.**  
(Orientador)

**Viviane Cátia Köhler, PhD.**  
(UFSM)

**Fabio Teixeira Franciscato, Msc.**  
(UFSM)

Santa Maria. 09 de dezembro de 2015

## **AGRADECIMENTOS**

Agradeço em primeiro lugar, a todos os professores pelos ensinamentos passados durante todos estes anos, em especial a professora Dra. Viviane Cátia Köhler pelas oportunidades recebidas e pela amizade.

Ao meu orientador, prof. Dr. Claiton Colvero, quero expressar minha gratidão, que acreditou em mim, partilhou as suas ideias, seus conhecimentos.

Aos meus colegas de curso minha gratidão pelo apoio constante e dedicação, em especial Jonathan Preuss, Alberto Kummer, Antonico Carlos Misssio, e Giuliano Lanes pelo grande apoio, pelos ensinamentos e por todos os dias que incomodei vocês.

Em especial a minha mãe Tereza Copatti e minha namorada Marcela Zart Arend que sempre acreditaram e incentivaram a minha conquista.

Obrigado a todos os colegas do Curso de Redes, pela convivência, amizades e companheirismo.

## RESUMO

Trabalho de Conclusão de Curso (TCC)  
Colégio Técnico Industrial De Santa Maria  
Curso Superior de Tecnologia em Redes de Computadores  
Universidade Federal de Santa Maria

### **MIDDLEWARE DE RASTREABILIDADE PARA TECNOLOGIAS DE REDES SEM FIO**

AUTOR: RAFAEL COPATTI TRINDADE

ORIENTADOR: CLAITON PEREIRA COLVERO

CO-ORIENTADOR: MIGUEL BAUERMANN BRASIL

DATA E LOCAL DA DEFESA: SANTA MARIA, 09 DE DEZEMBRO DE 2015.

Este trabalho apresenta o detalhamento das atividades realizadas durante o período de desenvolvimento de um *software* de interface universal (*middleware*) para utilização com diferentes tecnologias de identificação eletrônica sem fio. Inicialmente este *middleware* foi desenvolvido para permitir a utilização destas tecnologias em ambientes hospitalares e clínicos, com o objetivo de integrar os sistemas de controle de pacientes já existentes com as mais modernas técnicas de identificação eletrônica do mercado, de forma transparente e eficaz para qualquer formato de entrada de dados. Foram avaliados como candidatos aos sistemas de identificação de pacientes as tecnologias de códigos de barras, redes adaptativas com nós ativos no padrão *ZigBee IEEE 802.15.4* e identificadores eletrônicos por radiofrequência – *RFID* passivos *EPC Gen2*. Para a implementação do sistema e a realização dos ensaios em laboratório foi necessário o aprendizado e familiarização com as tecnologias utilizadas na identificação eletrônica, da mesma forma que foram estudadas diferentes plataformas de desenvolvimento de aplicativos para a criação do *middleware*. Este amplo conhecimento adquirido em diferentes áreas durante este período foi fundamental para o desenvolvimento da interface de integração de acordo com os objetivos e requisitos do projeto original, mantendo as características do mesmo. Destacam-se como requisitos a flexibilidade na escolha dos sistemas de entrada de dados, assim como a escalabilidade de tecnologias, sendo possível a fácil inserção de novos padrões ou sistemas que venham a ser demandados durante a sua utilização.

Palavras-chave: Banco de dados, *RFID*, Código de barras, *ZigBee*, *Middleware*.

## **ABSTRACT**

Completion Of Course Work  
Colégio Técnico Industrial De Santa Maria  
Curso Superior de Tecnologia em Redes de Computadores  
Universidade Federal de Santa Maria

### **UNIVERSAL MIDDLEWARE FOR WIRELESS TRACKING TECHNOLOGIES**

AUTHOR: RAFAEL COPATTI TRINDADE  
SUPERVISOR: CLAITON PEREIRA COLVERO  
CO-SUPERVISOR: MIGUEL BAUERMANN BRASIL  
DATE AND PLACE OF DEFENSE: SANTA MARIA, DECEMBER 09, 2015.

This work shows a breakdown of activities undertaken during the period of development of a universal interface software (middleware) for use with different wireless electronic identification technologies. Initially this middleware was designed to allow the use of these technologies in hospital and clinical settings, with the objective of integrating patients control systems existing with modern electronics market identification techniques, transparently and effectively to any format data entry. They were evaluated as candidates for identification of patients' bar code technology systems, adaptive systems with active nodes on the IEEE 802.15.4 ZigBee and electronic radiofrequency identifiers - passive RFID EPC Gen2. To system implementation and realization of laboratory tests was necessary learning and familiarization with the technologies used in electronic identification, just as we studied different application development platforms for the creation of middleware. This extensive knowledge acquired in different areas during this period was fundamental to the development of integration interface in accordance with the objectives and requirements of the original design, keeping characteristics. It stands out as requirements to flexibility in the choice of data entry systems and scalability technologies, making it possible for easy insertion of new standards or systems that may be demanded during use

Keywords: Database, RFID, Barcode, ZigBee, Middleware.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Código de barras criado por Woodland.....	16
Figura 2 – Modelo de código de barras criado por Laurer .....	17
Figura 3 – Modelo de um código de barras UPC e EAN. ....	18
Figura 4 – Princípio de operação de um sistema com código de barras.....	18
Figura 5 – Estrutura de dados utilizada no SSCC .....	20
Figura 6 – Representação de leitura dos dados em um sistema <i>RFID</i> . ....	22
Figura 7 – Sistema <i>RFID</i> com seus componentes de operação.....	24
Figura 8 – Princípio de operação de um sistema <i>RFID</i> . ....	24
Figura 9 – Componentes de um <i>Tag RFID</i> . ....	25
Figura 10 – Etiqueta <i>RFID</i> passiva com detalhe da antena e componentes internos. ....	26
Figura 11 – Representação típica de um numeral no padrão <i>EPC</i> . ....	28
Figura 12 – Protocolos de padronização para <i>RFID</i> . ....	30
Figura 13 – Estrutura mínima de operação de um sistema de identificação <i>RFID</i> .....	31
Figura 14 – Componentes típicos para um <i>middleware</i> de sistemas <i>RFID</i> . ....	32
Figura 15 – Representação de uma rede típica WPAN de dispositivos <i>ZigBee</i> . ....	33
Figura 17 – Configuração do <i>software ZXing</i> para leitura do código de barras. ....	38
Figura 16 – Ligações dos cabos para a operação do leitor <i>RFID</i> utilizado. ....	40
Figura 18 – Tela de configuração do <i>software</i> comercial distribuído pela <i>Digi – X-CTU</i> . ....	41
Figura 19 – Página do desenvolvedor do <i>software Aker Middleware</i> - Compatibilidades .....	41
Figura 20 – <i>Software Aker Middleware RFID</i> com indicação do leitor XR440 da Motorola..	42
Figura 21 – Terminal com os dados recebidos através do sistema <i>RFID</i> . ....	43
Figura 22 – Aplicativo demonstrando os dados recebidos do código de barras.....	43
Figura 23 – Terminal de visualização dos dados recebidos do sistema <i>ZigBee</i> . ....	43
Figura 24 – Modelo ER do relacionamento externo do banco de dados utilizado. ....	44



Figura 25 – Modelo ER do banco de dados utilizado no <i>middleware</i> .....	45
Figura 26 – Terminal de leitura dos dados obtidos do <i>transponder RFID</i> .....	47
Figura 27 – Dados recebidos no terminal de identificação <i>ZigBee</i> . ....	47
Figura 28 – Código para captura dos dados dos <i>tags RFID</i> utilizados.....	48
Figura 29 – Código utilizado para captura das informações do código de barras.....	49
Figura 30 – Exemplo de parte do código para leitura do <i>transponder ZigBee</i> .....	49
Figura 31 – Terminal de demonstração dos dados obtidos da leitura do <i>ZigBee</i> .....	50
Figura 32 – Código utilizado para executar o simulador da tecnologia <i>RFID</i> .....	50
Figura 33 – Tela de demonstração do ensaio de rastreabilidade executado.....	51
Figura 34 – Tabela referente as informações de rastreamento do paciente. ....	51
Figura 35 – Tela do terminal de leitura do código de barras.....	52
Figura 36 – Demonstração do <i>status</i> de entrada/saída do paciente com código de barras.....	52
Figura 37 – Tela do <i>middleware</i> com a leitura dos <i>transponders RFID</i> dos pacientes.....	53
Figura 38 – Arquivo de <i>log</i> demonstrando as informações relevantes do <i>frame API</i> .....	54
Figura 39 – Ensaio de operação do <i>middleware</i> realizados com os dispositivos <i>ZigBee</i> . ....	54

## LISTA DE TABELAS

Tabela 1 – Estrutura do identificador GTIN.....	19
Tabela 2 – Estrutura do identificador GLN.....	20
Tabela 3 – Distâncias típicas de leitura de acordo com a frequência <i>RFID</i> .....	25
Tabela 4 – Principais frequências utilizadas em sistemas <i>RFID</i> comerciais.....	27
Tabela 5 – Estrutura de classes <i>RFID</i> segundo o padrão <i>EPCglobal</i> .....	29
Tabela 6 – Especificações do leitor <i>RFID</i> XR400 da Motorola.....	39

## LISTA DE ABREVIATURAS E SIGLAS

<b>UHF</b>	<i>Ultra High Frequency</i>
<b>RFID</b>	<i>Radio Frequency Identification</i>
<b>ZIGBEE</b>	Tecnologia <i>WPAN</i> de rede sem fio
<b>WPAN</b>	<i>Wireless Personal Address Network</i>
<b>PAN ID</b>	<i>Personal Address Network Identification</i>
<b>GTIN</b>	Número Global do Item Comercial
<b>GLN</b>	Número Global de Localização
<b>SSC</b>	Código de série da unidade Logística
<b>GRA</b>	Identificador Global de Ativo Retornável
<b>GSRN</b>	Número Global de Relação de Serviço
<b>GIAI</b>	Identificador Global de Ativo Individual
<b>GDTI</b>	Identificador Global do Tipo de Documento
<b>GSIN</b>	Número Global de Identificação de Carga
<b>GINC</b>	Número Global de Identificação de Consignado
<b>TAG</b>	<i>Transponder RFID</i>
<b>EPC</b>	<i>Electronic Product Code</i>
<b>SQL</b>	<i>Structure Query Language</i>
<b>HTML</b>	<i>Hypertext Markup Language</i>
<b>HTTP</b>	<i>HyperText Transfer Protocol</i>
<b>URL</b>	<i>Uniform Resource Locator</i>
<b>API</b>	<i>Application Programming Interface</i>
<b>UFSM</b>	Universidade Federal de Santa Maria
<b>CTISM</b>	Colégio Técnico Industrial de Santa Maria

## SUMÁRIO

INTRODUÇÃO.....	13
1. REVISÃO BIBLIOGRÁFICA.....	15
1.1. Código de barras.....	16
1.1.1. Princípio de funcionamento do código de barras .....	17
1.1.2. Chaves de identificação de códigos de barras .....	19
1.1.3. Vantagens e desvantagens dos códigos de barras.....	21
1.2. <i>RFID</i> - Identificação por radiofrequência. ....	21
1.2.1. Componentes de um sistema <i>RFID</i> .....	22
1.2.1.1. Leitores <i>RFID</i> .....	24
1.2.1.2. <i>Tag RFID</i> .....	25
1.2.2. Frequências de Funcionamento da Tecnologia <i>RFID</i> .....	27
1.2.2.1. <i>EPCglobal Network</i> .....	28
1.2.2.2. <i>EPCglobal Gen2</i> .....	29
1.2.3. <i>Middleware</i> para o sistema <i>RFID</i> .....	30
1.2.3.1. Componentes para o <i>middleware RFID</i> .....	31
1.3. Padrão <i>ZigBee</i> .....	32
1.3.1. Atribuição dos dispositivos <i>ZigBee</i> .....	33
1.4. Topologia da rede padrão <i>ZigBee</i> .....	33
1.5. Banco de dados <i>MYSQL</i> .....	34
1.6. Linguagem de programação <i>PHP</i> .....	34
2. MÉTODOS E TÉCNICAS.....	35
2.1. Ambiente de desenvolvimento .....	36
2.1.1. Tecnologias e <i>hardwares</i> .....	37
2.1.1.1. Leitores de código de barras.....	37
2.1.1.2. Leitores <i>RFID</i> .....	38
2.1.1.3. Dispositivos <i>ZigBee</i> .....	40
2.2. <i>Middleware</i> .....	41
2.3. Banco de Dados.....	44
2.4. Implementação do <i>software</i> .....	46
2.5. Ensaio para obtenção das informações dos leitores das tecnologias .....	46
3. RESULTADOS E DISCUSSÃO .....	48
CONCLUSÃO.....	55
Sugestões para trabalhos futuros. ....	56
REFERÊNCIAS .....	57

## INTRODUÇÃO

Embora o ambiente hospitalar não faça parte do cotidiano da maioria das pessoas, ninguém está livre de algum dia ter que utilizar os serviços prestados pelo sistema de saúde público ou privado. Em geral, quem recorre aos serviços descritos anteriormente encontra-se em uma situação delicada, envolvendo não somente as condições físicas da própria doença em questão, mas também o lado emocional e psicológico da pessoa, que pode estar fortemente abalado. Nestas condições, os pacientes procuram um atendimento dos prestadores de serviços de saúde de qualidade, com a maior eficiência possível e principalmente com profissionalismo e cordialidade no encaminhamento dos processos.

É importante observar que muitos dos pacientes atendidos sentem-se incomodados com o próprio ambiente, em diferentes formas de expressão, como inquietude, tensão, nervosismo e até mesmo em alguns casos com extrema agressividade ou incompreensão dos fatos apresentados, justificável pela fragilidade emocional de muitos nestes momentos.

Devido ao excessivo número de pacientes que diariamente são submetidos a procedimentos, tratamentos, exames e demais serviços hospitalares, o monitoramento e controle preciso destes usuários se torna uma tarefa desafiadora, se considerar as diversas variáveis envolvidas neste processo de atendimento, que por ser seriado inclusive, resultando em fortes dependências com outros atendimentos na fila de espera. Desta forma, foi desenvolvido um estudo para detecção e possível correção de alguns pontos de falhas no sistema atual de fluxo de atendimento de pacientes, considerando a grande maioria dos prestadores de serviço, que nestes processos costumam utilizar procedimentos manuais.

Neste trabalho foi observado que os principais problemas relatados em hospitais e clínicas são: tempo de espera pré e pós atendimento, local de espera inadequado e falta de informações básicas dos procedimentos realizados. Levando-se em consideração que pelas fragilidades dos pacientes descritas anteriormente, estes fatores contribuem para que os problemas aparentem possuir uma gravidade maior do que realmente apresentam. Desta forma, foi possível verificar que em geral clínicas e hospitais possuem um problema sério de gerenciamento do seu ambiente e de seus pacientes, mas que pode ser resolvido em grande parte através da adoção de novas tecnologias de comunicação e identificação.

Para amenizar estes impactos aos usuários do sistema de saúde, foram analisadas as tecnologias de identificação e redes de comunicação dentro de uma ideia preliminar de

desenvolver um comparativo entre as três tecnologias de identificação eletrônica mais conhecidas com *RFID*, *ZigBee* e código de barras. Com base nos resultados poderia ser realizada a seleção da tecnologia que melhor poderia automatizar os processos de identificação, procedimentos atendimento e rastreabilidade de pacientes dentro de hospitais e clínicas.

Na busca por soluções para utilizar as tecnologias empregadas no projeto notou-se a falta de um *software* para que fosse capaz de interagir com diferentes tecnologias para rastreabilidade. Desta forma, foi alterado o escopo original para o desenvolvimento de um *software (middleware)* universal, com a capacidade de suportar todas as tecnologias apresentadas através de uma única interface padronizada para os dados.

O objetivo principal deste trabalho de conclusão de curso foi direcionado para o desenvolvimento um *middleware* universal para integração de sistemas de gerenciamento de localização de indivíduos em ambientes fechados. Este *software* também é responsável pelo tratamento dos dados de forma adequada e transparente aos usuários e sistemas de controle específicos e proprietários de ambientes hospitalares.

O *middleware* desenvolvido apresenta um suporte para integração dos sistemas convencionais de cadastro de pacientes já utilizados nestes locais com as diferentes tecnologias de identificação eletrônica, como por exemplo, a tecnologia de identificação por radiofrequência - *RFID*, o código de barras e o *ZigBee*, embora possa ser expandido para quaisquer outras tecnologias adicionais.

Para atingir o objetivo proposto de forma satisfatória fez-se necessário descobrir os recursos mínimos necessários para o bom funcionamento da rede; desenvolver uma ferramenta de *software* preliminar para a comunicação dos dispositivos *RFID*, Código de barras e *ZigBee* com o servidor, e com funcionalidades de tratamento das informações recebidas através da análise dos dados de operação dentro de um ambiente de testes controlado.

O projeto foi desenvolvido em três etapas distintas, que correspondem à aquisição de conhecimentos sobre as tecnologias existentes de identificação eletrônica, através de pesquisas bibliográficas e testes em laboratório, a elaboração do *software* de interface universal através das ferramentas de programação disponíveis, e por fim a realização de ensaios em ambiente controlado para verificação do funcionamento do sistema e realização de possíveis correções.

## 1. REVISÃO BIBLIOGRÁFICA

Para a especificação das interfaces do *middleware* proposto, com objetivo de tornar ele universal para as tecnologias de identificação eletrônica, foram separadas três possibilidades de baixo custo que atendem as demandas com suas restrições, sendo o código de barras óptico, a identificação por radiofrequência – *RFID* com *transponders* passivos e a identificação de nós de redes industriais adaptativas e sem fio no padrão comercial *ZigBee*.

Como este trabalho é direcionado ao atendimento em ambientes de clínicas e hospitais, a rastreabilidade passa a ser então um caminho para que estas instituições implementem sistemas de gerenciamento, propiciando a eficiência em suas operações. Essas tecnologias aplicadas à logística do processo de rastreabilidade, procedimentos e atendimento diferenciado aos pacientes, objetivam minimizar tempo de espera para procedimentos e evitar erros, agilizando os processos que são executados manualmente. Atualmente, a tecnologia que é mais utilizada para identificação de pacientes no Brasil utiliza a códigos de barras. Entretanto, como em hospitais e clínicas existe um grande volume de pacientes esta tecnologia demanda leitura de cada usuário através do escaneamento individual, o que não permite processamento rápido e automático, ao mesmo tempo que aumenta a possibilidade de erros na leitura. Além disto, não existe uma rede de rastreabilidade que permita a manutenção do histórico de cada paciente, seja em tempo real, ou até mesmo durante o seu período de permanência no local.

Tecnologias de identificação mais modernas, como a identificação por radiofrequência (*RFID*) vem sendo estudadas e desenvolvidas como soluções a serem utilizadas na área de saúde, além de outros setores de atividade em que a tecnologia já foi reconhecida como eficiente, como na automação dos processos de logística, controle e de circulação de mercadorias (Perin, 2012). Diferentes dispositivos e redes de identificação e comunicação sem fio podem ser implementadas de forma a suprir as demandas do projeto, como é o caso da utilização do *ZigBee*, que agrega novas funcionalidades ao sistema. As áreas que mais são beneficiadas por soluções de rastreabilidade com *RFID* e *ZigBee* em hospitais e clínicas são a da identificação de pessoas, monitoramento de equipamentos e controle de ativos. Deste modo observa-se que uma solução para monitoramento e controle de paciente baseado nestas tecnologias contribuirá para a gestão hospitalar e de clínicas, oferecendo agilidade e automatização nos processos, aumentando a eficiência, diminuindo tempo de espera e diminuição de erros relacionados, aos sistemas manuais, onde possa ocorrer o erro humano.

Com base nestas possibilidades, as diferentes tecnologias e ferramentas de desenvolvimento estão sendo estudadas e brevemente descritas neste capítulo.

### 1.1. Código de barras

O código de barras é uma representação de símbolos que os traduz em números para representarem produtos, serviços, documentos, entre outros, facilitando a captura dos dados através de leitores ópticos e procedimentos de automação de processos.

Em 1948 Joseph Woodland pensando em como criar um modo de automatizar os processos de identificação de produtos desenvolveu um símbolo simples em que no momento que fosse escaneado, teria suas informações transcritas em sequências numéricas, que processadas pelo computador, efetuaría uma referência do código a um produto específico.

A Figura 1 demonstra o primeiro modelo de código de barras que foi patenteado em 1952 por Joseph Woodland e Bernard Silver, mas pelas limitações de que escâneres e outros equipamentos teriam custo muito caro para colocar o sistema para operar, este projeto ficou parado por 20 anos (BARCODING, 2010).



Figura 1 – Código de barras criado por Woodland.  
Fonte: MARS (2014).



Somente na década de 70, George Laurer foi incumbido de colocar em prática o sistema de Woodland. Mas Laurer achava que neste sistema faltava algo para suprir as necessidades do setor varejista, foi então que Laurer criou símbolo em formato retangular que armazenava mais informações. O código criado por Laurer recebeu nome de UPC (*Universal Product Code*) e está representado na Figura 2. Em 1974 o primeiro produto a utilizar o código de barras foi uma caixa de chicletes Wrigley, desta forma dando início ao processo de utilização comercial deste padrão, que hoje é presente em 99% dos produtos (MARS,2014).



Figura 2 – Modelo de código de barras criado por Laurer  
Fonte: MARS (2014).

### 1.1.1. Princípio de funcionamento do código de barras

O código de barras é uma tecnologia de identificação automática utilizada mundialmente por muitas áreas e atividades, sejam elas para controle de mercadorias, pessoas, objetos, patrimônios e rastreamento. Para que o sistema seja utilizado em âmbito mundial, existem entidades que contribuem para as especificações do código de barras. Para isto é preciso que haja uma padronização dos modelos adotados por estas entidades. As entidades que contribuíram para que o padrão (INFO, 2015) seja usado globalmente são:

A UPCC (*Uniform Product Code Council*) que estabeleceu as normas para a UPC (Código Universal de Produto) na identificação de produtos e das comunicações eletrônicas aplicadas, que posteriormente deu origem a UCC (*Uniform Code Council*), a AIM (*Association for Automatic Identification and Mobility*) e EAN (*European Article Numbering*).

Em 2005 após a fusão da UCC e a EAN deu origem a nova entidade, a GS1, que gere o uso global dos sistemas baseados em código de barras conhecidos GTIN.

Em geral os códigos de barras utilizam dois tipos de tecnologias:

Tecnologia Linear, que são reconhecíveis e utilizadas no mundo todo, onde utiliza barras verticais com diferentes larguras separadas por espaços em branco; e a Tecnologia

Bidimensional ou 2D, que utiliza barras, símbolos, quadrados e pontos. Na Figura 3 encontra-se a representação de um exemplo de código de barras normatizado pela GS1.

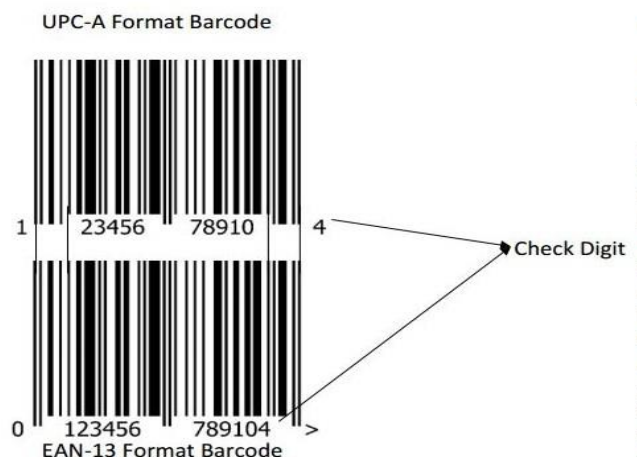


Figura 3 – Modelo de um código de barras UPC e EAN.  
Fonte: GS1 BRASIL (2015).

O código de barras UPC é utilizado no varejo dos EUA (Estados Unidos) e Canadá, utilizando o identificador GTIN-12 de doze caracteres para identificação. Já o código EAN é parecido com os códigos UPC, mas sendo que estes têm 13 dígitos.

O código de barras funciona praticamente como números binários onde espaços em branco equivalem a "0" e barras equivalem a "1", sendo que o mesmo contém as informações de prefixo do país de registro da empresa, a identificação da empresa, referência do produto e dígito verificador (INFO, 2015). Na Figura 4 pode-se observar um exemplo da identificação de parâmetros através de um código de barras convencional.



Figura 4 – Princípio de operação de um sistema com código de barras.  
Fonte: MARS (2014).

### 1.1.2. Chaves de identificação de códigos de barras

As chaves identificadoras têm como objetivo identificar com exatidão as informações comerciais, em qualquer setor ou país em escala mundial. Este sistema de chaves identificadoras são utilizadas para recuperar informações pré-definidas de qualquer produto ou processo, sendo que as empresas podem atribuir identificação exclusiva (GS1BR,2015).

Os identificadores mais utilizados são o GTIN (Número Global do Item Comercial), o GLN (Número Global de Localização) e o SSCC (Código de série da unidade Logística).

O GTIN é um identificador para itens comerciais, embora também possam ser atribuídos para quaisquer outros itens, produtos ou serviços que se necessite obter informações adicionais de forma eletrônica. O GTIN recupera as informações pré-definidas e armazenada na base de dados do sistema informatizado, para processar a transação pretendida posteriormente. Existem 4 modelos de GTIN disponíveis no mercado, cada um com tamanho diferente e compreendidos com 8, 12, 13 e 14 dígitos, dependendo da aplicação que se necessite utilizar.

O GTIN-8 é utilizado no código de barras EAN-8, já o GTIN-12 é mais utilizado no código de barras UPC-A, o GTIN-13 é codificado no EAN-13 e o GTIN-14 no ITF-14. (GS1BR, 2015). Na Tabela 1 pode-se verificar os exemplos da estrutura de dados GTIN de acordo com o número de dígitos da tecnologia escolhida.

Tabela 1 – Estrutura do identificador GTIN.

GTIN - Global Trade Item Number (14 dígitos)														
Estrutura de Dados	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14
GTIN-14	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14
GTIN-13	0	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13
GTIN-12	0	0	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12
GTIN-8	0	0	0											

Fonte: GS1 BRASIL (2015).

O GLN é conhecido como o número global de localização, possuindo um tamanho de 13 dígitos. Este número pode estar vinculado aos dados da empresa, localizações físicas ou entidades associadas a GS1. Um dos benefícios do uso de GLN é que ele fornece um padrão interno e externo para identificação de locais (GS1BR,2015). Na Tabela 2 está sendo apresentada a estrutura dos dados contidos no identificador GTIN.

Tabela 2 – Estrutura do identificador GLN.

Prefixo GS1 de Empresa / Referência da Localização												Dígito Verificador
N <sub>1</sub>	N <sub>2</sub>	N <sub>3</sub>	N <sub>4</sub>	N <sub>5</sub>	N <sub>6</sub>	N <sub>7</sub>	N <sub>8</sub>	N <sub>9</sub>	N <sub>10</sub>	N <sub>11</sub>	N <sub>12</sub>	N <sub>13</sub>

Fonte: GS1 BRASIL (2015).

Por último, o SSCC é conhecido como identificador de chave, sendo composto por 18 dígitos que identificam as informações sobre o produto e o lote, desta forma simplificando o processo de rastreabilidade. Na Figura 5 pode-se observar a estrutura de dados do SSCC definido pelo GS1, com demonstração das características de informações de origem.



Figura 5 – Estrutura de dados utilizada no SSCC

Fonte: GS1 BRASIL (2015).

Para garantir a integridade dos dados na leitura o código utiliza um dígito verificador. Este controle é realizado em função dos números anteriores, sendo utilizado por todas as estruturas GTINS. A escolha do padrão ideal depende da natureza do produto e sua aplicação. Após escolha da estrutura de identificação, é atribuído seu número de produto e o dígito verificador, que é o último dígito da sequência de números. Ele é utilizado para verificar possíveis erros na codificação ou na digitalização deste.

Para calcular o dígito verificador segue-se os seguintes passos (GS1BRASIL, 2015b):

- Numera-se da direita para esquerda;
- Em todos os dígitos pares atribui-se o valor 1;
- Em todos os dígitos ímpares atribui-se o valor 3;
- Multiplicam-se os valores e atribui-se ao número;
- Faz-se o somatório dos valores já multiplicados;
- Busca-se o múltiplo de 10 mais próximo da soma dos produtos, que seja maior ou igual a essa soma;
- Subtrai-se o somatório do múltiplo de 10.

### 1.1.3. Vantagens e desvantagens dos códigos de barras

Na breve descrição do funcionamento do código de barras, notou-se o grande impacto que esta tecnologia teve no mercado mundial. Esta tecnologia ao ser implantada principalmente no setor varejista, contribui indiscutivelmente no ganho de produtividade e agilidade em processos que eram realizados manualmente, diminuindo o tempo de identificação de produtos e serviços. Como principais vantagens pode-se citar as seguintes:

- Baixo custo de implementação e manutenção;
- Tecnologia simples e barata;
- Aumento da produtividade;
- Tecnologia estabilizada;
- Utilização da infraestrutura existente;
- Confiabilidade do sistema;
- Imune a interferências eletromagnéticas;

Como qualquer tecnologia no mercado, nenhuma está imune a problemas de vulnerabilidades e limitações, sendo suas principais desvantagens:

- De fácil falsificação, pois não existe nenhum tipo de segurança;
- É sensível ao material que é impressa;
- Distância para leitura do código de barras;
- Ambiente úmido pode danificar e interferir na leitura;
- Leem um código por vez;
- Leitura manual e individual;
- Pode ser danificada facilmente, com caneta, estilete, rasgada, entre outros.

### 1.2. *RFID* - Identificação por radiofrequência.

De acordo com Pinheiro o *RFID* é uma tecnologia sem fio que utiliza radiofrequência para identificação automática de um objeto. Pode-se armazenar e recuperar informações que estão referenciados a ele. Os objetos são etiquetados com uma *tag*, que identifica este objeto utilizando ondas de rádio, e um leitor de *RFID* captura automaticamente os dados contidos nesta *tag*, sendo estas processadas pelo sistema (PINHEIRO, 2006).

A tecnologia *RFID* é utilizada para rastrear e gerenciar produtos, documentos e serviços. Por ela ser uma tecnologia de comunicação por radiofrequência, não necessita de contato físico com leitores nem de campo visual, como é fundamental nos códigos de barras. Estes últimos necessitam de visada direta para poderem ser lidos. Na Figura 6 está sendo apresentado um exemplo do funcionamento para a leitura dos dados enviado pelo *transponder*.

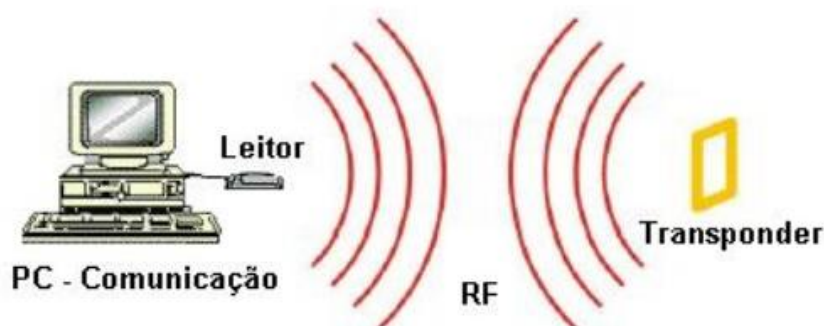


Figura 6 – Representação de leitura dos dados em um sistema *RFID*.  
Fonte: SANTOS E PINHEIRO (2006).

A origem da tecnologia *RFID* se deu na Segunda Guerra Mundial, onde eram utilizados radares que foram inventados em 1935 pelo físico Robert Alexander Watson Watt. Eles podiam avisar com antecedência a aproximação de aviões através da captura do sinal de um avião estava se aproximando, sem poder identificar se era aliado ou inimigo. Watt desenvolveu o primeiro identificador ativo de amigos ou inimigos IFF (Identificador de amigos ou inimigos), e desta forma todo avião britânico recebeu um identificador que funcionava como um transmissor, que ao receber os sinais do radar, transmitiam um sinal de resposta.

O primeiro sistema comercial a utilizar etiquetas *RFID* foi o de antifurto, onde identifica-se se um item foi roubado. A partir disto a tecnologia tem sido amplamente estudada para ser utilizada em outras áreas, desde sistemas simples de controle de estoque e empréstimos em bibliotecas, até a rastreabilidade de equipamento e indivíduos. (IDESEG, 2015).

### 1.2.1. Componentes de um sistema *RFID*

Um sistema *RFID* é composto basicamente por leitores, *tag* ou *transponders*, antenas e *enterprise subsystem* (*middleware*, sistemas analíticos, infraestrutura de rede), e desta forma estão descritos de acordo com suas características:

- Leitor *RFID* (*Transceiver*): Responsável pela emissão dos sinais de radiofrequência, que é a fonte de energia que alimenta a *tag RFID* passiva. Esta *tag* em especial após ser alimentada responde com os dados que estão em sua memória, diferente de um leitor de código de barras que precisa ter contato visual entre leitor e código. O que diferencia o sistema *RFID* é que ele não precisa ter o contato físico ou visual. Existem diferentes leitores para diferentes *tags*, como ativas ou passivas, tamanho da antena, frequência utilizada, entre outras características (PHILLIPS, 2007).
- *Tag RFID* ou (*Transponder*): são dispositivo de identificação com ID único utilizados para identificação de produtos, objetos, serviços ou rastreamento. Elas podem ser *tags* ativas, que utilizam bateria para obter energia necessária para comunicar-se com leitor, ou passivas, sem bateria, que utilizam a radiofrequência para obter energia necessária para comunicações com leitor.
- Antenas *RFID*: estão conectadas ao dispositivo de leitura para captação de dados e envio do sinal para que as *tags* sejam lidas.
- *Enterprise Subsystem*: é composto pelo *middleware*, sistema analítico e infraestrutura de rede. Ele é responsável por interligar os diversos equipamentos que compõem o sistema de identificação *RFID*.
- *Middleware*: é o responsável pela conexão e tratamento das informações entre o sistema de gestão e do sistema específico de *RFID*, que é composto pelos leitores, *tags* e infraestrutura de redes de comunicação.
- Sistemas Analíticos: são compostos de bases de dados, aplicações de processamento de dados e servidores *web*, que processam as saídas de dados do *middleware* baseado em requisitos de aplicações e instruções de utilização.
- Infraestrutura de Rede: permite a comunicação entre os subsistemas de radiofrequência utilizados para a identificação eletrônica dos transponders *RFID* e os subsistemas de gestão das informações e recursos, responsáveis pela operação.

Na Figura 7 está representado o funcionamento de um sistema *RFID* com os seus respectivos componentes que compõem uma estrutura necessária básica de operação, para que o sistema externo possa interagir com sistema *RFID*. Nota-se nesta imagem a representação dos elementos que formam o *enterprise subsystem*, que é responsável por interligar os diferentes sistemas que integram a tecnologia *RFID*.

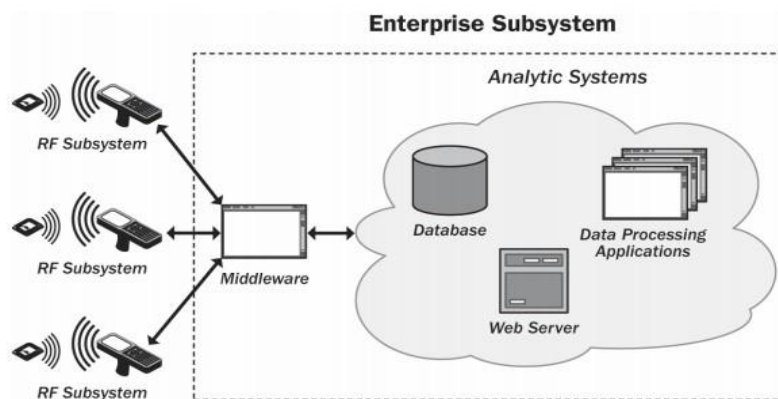


Figura 7 – Sistema *RFID* com seus componentes de operação.  
Fonte: PHILLIPS (2007).

### 1.2.1.1. Leitores *RFID*

Segundo Glover e Bhatt o leitor *RFID* (*transceiver*) é um sistema de comunicação entre a tecnologia *RFID* e o sistema gerente, onde os dados são processados. Estes leitores são responsáveis por emitir as ondas de radiofrequência para alimentar as *tags* passivas que estejam ao seu alcance. Desta forma, as *tags* recebem energia suficiente para trocar informações com o leitor, enviando os dados que estão gravados para que estes sejam tratados pelo *middleware*, e posteriormente processados pelo sistema (GLOVER; BHATT, 2006).

Na Figura 8 pode-se observar um exemplo do funcionamento do sistema para leitura das *tags RFID*, onde as antenas estão localizadas em pontos específicos para fazer a identificação, enviando os dados ao *middleware*, que faz o filtro destes (GLOVER; BHATT, 2006).

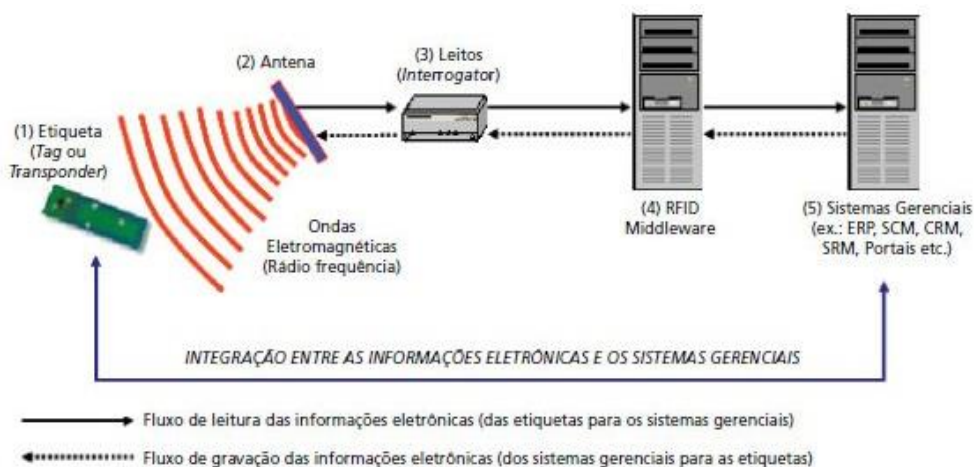


Figura 8 – Princípio de operação de um sistema *RFID*.  
Fonte: IBM (2013).



### 1.2.1.2. Tag RFID

A *tag* (*transponder* ou etiqueta inteligente) é responsável pela identificação do sistema *RFID*. Segundo Santini, as *tags* são responsáveis por armazenar, enviar e transmitir os dados, sendo a *tag* um dos componentes principais nestes sistemas, por ela fazer a identificação única dos elementos a serem rastreados (SANTINI, 2008). O dispositivo é composto por três componentes básicos: antena, circuito integrado e encapsulamento. Desta forma os componentes descritos constituem a *tag* onde são guardadas as informações. Na Figura 9 pode-se observar os componentes que constituem a *tag* (GLOVER; BHATT, 2006).

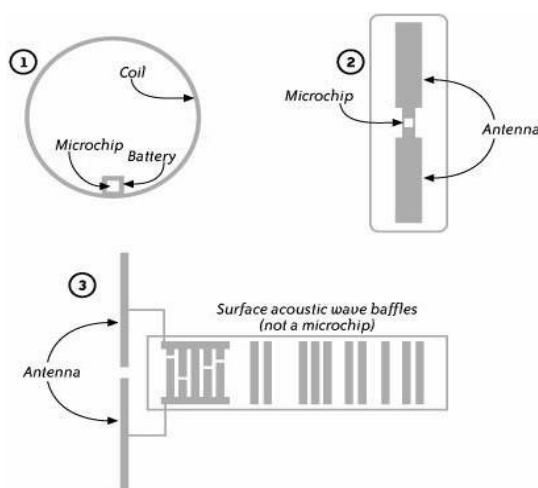


Figura 9 – Componentes de um *Tag RFID*.

Fonte: GLOVER AND BHATT (2006).

De acordo com a sua construção, são definidas as faixas de frequências de operação e o alcance de leitura máximo, conforme está demonstrado na Tabela 3.

Tabela 3 – Distâncias típicas de leitura de acordo com a frequência *RFID*.

Frequência	Faixa máxima típica para identificadores passivos	Algumas aplicações típicas
LF	50 centímetros	Identificação de animais de estimação e leituras próximas de itens com alto conteúdo de água
HF	3 metros	Controle de acesso a ambientes
UHF	9 metros	Caixas e embalagens
Microondas	> 10 metros	Identificação de veículos de todos os tipos

Fonte: GLOVER AND BHATT (2006)

A *tag* ao entrar no raio de leitura de um *transceiver* é induzida a responder as solicitações de leitura realizadas pelo leitor (ALMEIDA, 2011). As *tags* são divididas em três grupos sendo eles: ativos, passivas e semi-passiva.

As *tags* passivas segundo a IBM não possuem fonte de energia própria, não utilizam bateria interna e dependem da alimentação que é fornecida pelo *transceiver* para comunicar-se. Estas *tags* respondem a uma excitação que é gerada pelo envio do sinal do *transceiver* e transmitem as informações armazenadas na sua memória (IBM, 2013).

Segundo Gomes as *tags* podem ter uma vida longa sem precisar de manutenção pela ausência de bateria, além de suportar condições extremas de ambientes. O baixo custo de produção e de comercialização, leva as *tags* passivas serem as mais utilizadas na implementação dos sistemas *RFID* (GOMES, 2007). Estas *tags* podem ser somente leitura (*read only*) ou leitura e escrita (*read/write*). Na Figura 10 está sendo demonstrado um exemplo de *tag* passiva que necessita de excitação para responder ao leitor *RFID*.

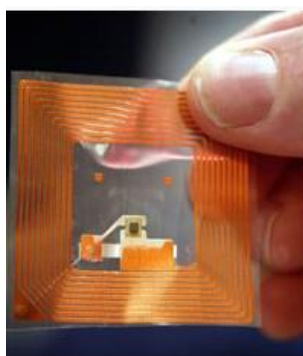


Figura 10 – Etiqueta *RFID* passiva com detalhe da antena e componentes internos.  
Fonte: IBM (2013).

Ainda segundo a IBM, as *tags* ativas possuem uma fonte interna de alimentação (bateria) que fornece energia ao seu circuito integrado, e utiliza esta energia para comunicar-se com o leitor, sem a necessidade de utilizar radiofrequência para excitar-se. No geral, as *tags* ativas permitem escrita e leitura (IBM, 2013). Em comparação das *tags* ativas com as passivas, estas possuem um custo elevado, são maiores, tem maior alcance e o tempo de vida útil depende da bateria. Alguns modelos suportam a troca de baterias, em outros as *tags* são seladas, sendo assim sem suporte a troca de bateria.

A terceira classificação corresponde as *tags* semi-passivas, que possuem uma fonte interna de alimentação, e segundo a IBM a mesma fornece energia necessária para alimentar os

circuitos internos, mas não utiliza esta energização para comunicação com o leitor. Como as características são semelhantes das *tags* passivas, ela depende do sinal do leitor para fazer sua comunicação entre os dispositivos (IBM, 2013). As *tags* permanecem em modo desligado até o momento que recebem um sinal das ondas eletromagnéticas do leitor. Deste modo o leitor excita a *tag* que está em seu campo de leitura e esta responde com suas informações gravadas.

### 1.2.2. Frequências de Funcionamento da Tecnologia *RFID*

As principais frequências do *RFID* utilizadas para aplicações são:

- *Low Frequency* (Baixa Frequência - LF): faixa de frequências de 30 kHz até 300 kHz. As *tags* desta faixa de frequências operam na faixa de 125 kHz ou 134,2 kHz e são geralmente etiquetas passivas, com uma distância de leitura de alguns centímetros.

- *High Frequency* (Alta Frequência - HF): faixa de frequências de 3 MHz até 30 MHz. As *tags* construídas para 13,56 MHz são normalmente utilizadas como crachás para identificação individual. Podem ser utilizadas para identificar objetos individuais, e a distância de leitura chega a ser maior do que 10 cm.

- *Ultra High Frequency* (Ultra Alta Frequência - UHF): faixa de frequências de 300 MHz até 1 GHz. Nesta faixa de frequências os *transponders* podem ser ativos ou passivos. Existe uma faixa bastante utilizada que é a de 868 MHz na Europa e de 915 MHz nos Estados Unidos e Brasil. Essas *tags* são utilizadas também em processos de rastreamento de ativos ou produtos em lojas, controle de estoque, inventários e identificação veicular.

Na Tabela 4 pode-se observar uma comparação entre as faixas de frequências comerciais dos sistemas de identificação e suas principais utilizações.

Tabela 4 – Principais frequências utilizadas em sistemas *RFID* comerciais.

Frequência	Nome	Aplicações
135 kHz	<i>Low Frequency</i> (LF)	Controle de Acesso, Smart Cards
13.553 13.567 MHz	<i>High Frequency</i> (HF)	Smart Cards
860 - 960 MHz	<i>Ultra High Frequency</i> (UHF)	Smart Labels
2.446 - 2.454 GHz	<i>Super High Frequency</i> (SHF)	Identificação de Veículos

Fonte: GLOVER AND BHATT (2006)

### 1.2.2.1. EPCglobal Network

A *EPC* (*Electronic Product Code*) é uma entidade sem fins lucrativos criada para desenvolver padrões baseado em especificações *EPC* (*Electronic Product Code*), tendo em vista promover a adoção desta padronização mundial para identificação automática de qualquer item em qualquer localização geográfica. O *EPC* foi desenvolvido com intuito de substituir o código de barras, atribuindo a cada objeto um identificador único e informações de rastreabilidade. A estrutura básica do padrão *EPC* é dividida em 4 áreas, sendo:

- A primeira parte identifica a versão do código do produto, informando quantos bits possui;
- A segunda parte é a identificação do fabricante;
- A terceira parte é o identificador da categoria do produto;
- E a última e quarta parte é o número serial de cada produto especificamente.

Na Figura 11 pode-se observar a representação de um exemplo da construção do formato do número *EPC* para um determinado item arbitrário.



Figura 11 – Representação típica de um numeral no padrão *EPC*.  
Fonte: RFID POINT (2015).

A *EPCglobal* além de adotar os padrões da numeração de identificação para as *tags*, criou adicionalmente uma classificação de acordo com suas tecnologias:

- As *tags* com classificação 0 e 1 permitem somente a leitura e são passivas.
- As *tags* com classificação 2 permitem a leitura e a escrita, e são passivas com a mesma funcionalidade da classe 1, mas com maior memória e implementação de criptografia.
- As *tags* com classificação 3 permitem a leitura e a escrita, mas utilizam bateria para alimentação. São do tipo semi-passiva.
- As *tags* com classificação 4 permitem a leitura e a escrita, utilizam bateria para alimentação, sendo do tipo ativas e podendo se comunicar com outras *tags*.

Na Tabela 5 estão exemplificadas as classificações que podem ser utilizadas e suas funcionalidades de acordo com cada classe do sistema *RFID EPCglobal*.

Tabela 5 – Estrutura de classes *RFID* segundo o padrão *EPCglobal*.

<b>Descrição da estrutura das classes <i>EPCglobal</i> definida para a tecnologia <i>RFID</i></b>	
<b>Classes</b>	<b>Funcionalidades</b>
Classe 0 / Classe 1	Etiquetas passivas para identificação
Classe 2	Etiquetas passivas com a mesma funcionalidade da classe 1, com maior memória e criptografia
Classe 3	Etiquetas semi-passivas com a mesma funcionalidade das etiquetas classe 1 e classe 2 + bateria + comunicação de banda larga
Classe 4	Etiquetas ativas com as mesmas funcionalidades das classes anteriores + comunicação <i>peer-to-peer</i> em banda larga com outras etiquetas ou leitores de dados

Fonte: RFID POINT (2015)

#### **1.2.2.2. *EPCglobal Gen2***

Inicialmente, os protocolos ISO 18000 e o protocolo *EPC* discordavam um do outro. Contudo, recentemente a *EPCglobal* sujeitou a primeira geração do seu protocolo Classe 1 para integrar ao protocolo ISO. Esse novo padrão, chamado *EPCglobal Gen2* (segunda geração do ECP Classe 1), é similar ao padrão ISO 18000-6 tipo C. Isso significa que os dispositivos que estão dentro da norma ISO 18000-6, também estão de acordo com o padrão *EPC Gen2* (INSTRUMENTS, 2011).

O maior problema antes deste novo padrão ser adotado, era de que as leitoras não conseguiam ler várias *tags* ao mesmo tempo. Após adotar o padrão *EPC Gen2*, um fator que melhorou foi a distância de leitura, que pode chegar até 10 metros. Também foi realizada uma pequena implementação de segurança e uma implementação das sessões, pois isso faz com que mais de um leitor possa comunicar-se com apenas uma *tag* ao mesmo tempo, sem que haja interferência entre eles. Deste modo o protocolo atual trabalha com uma densidade de leitura muito alta, sendo que pode ler milhares de *tags* por segundos. O protocolo Classe 1 *Gen2*, opera nas frequências UHF (860 até 930 MHz), de acordo com a regulamentação específica de cada país (RFIDPOINT, 2015).

A Figura 12 apresenta uma representação das camadas dos protocolos que formam as recomendações de padronização dos sistemas *RFID* modernos.

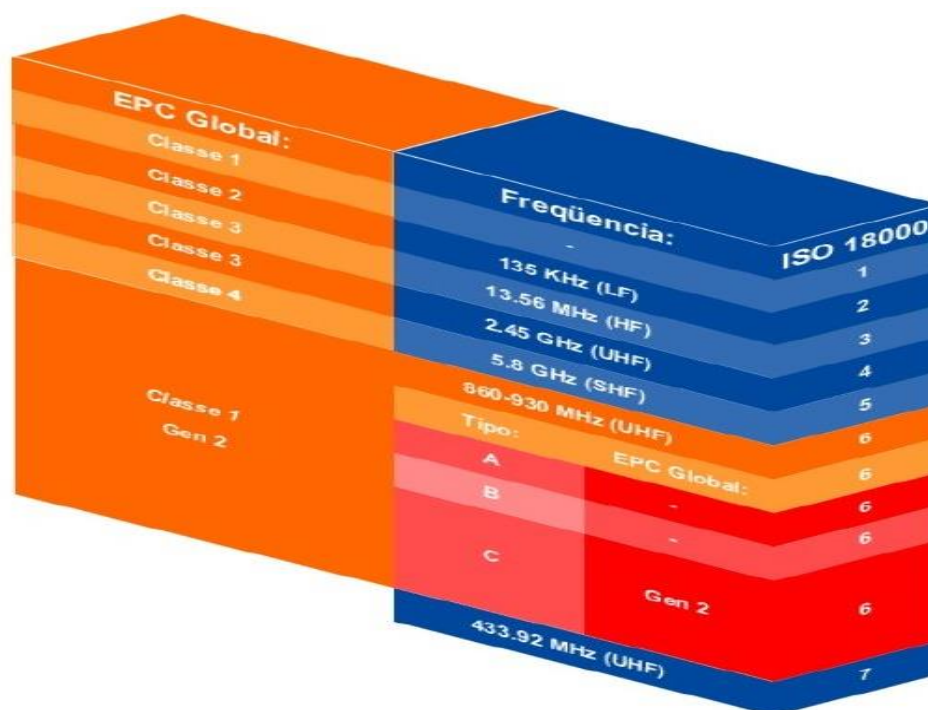


Figura 12 – Protocolos de padronização para *RFID*  
 Fonte: RFID POINT (2015).

### 1.2.3. *Middleware* para o sistema *RFID*

O *middleware* segundo Glover é quem faz a coleta, o tratamento, a filtragem de dados e agrega a identificação das *tags* que o leitor *RFID* identificou. O *middleware* é uma interface entre os leitores e o sistema de gerenciamento da tecnologia *RFID* (GLOVER; BHATT,2008).

Glover afirma que o *middleware* tem uma grande importância para a estrutura de sistemas *RFID*, por possibilitar absorção de diferentes tecnologias de leitores, desta forma integrando seus dados de maneira flexível (GLOVER; BHATT,2006). Freitas diz que nem todos sistemas *RFID* necessitam de um *middleware*, a implementação deste é necessária quando há um número alto de requisições das informações contidas nas etiquetas. Quando a disponibilidade do serviço tem que ser alta é devido a necessidade de manter um tráfego intenso de dados (FREITAS DIAS, 2012). Os componentes principais de uma infraestrutura comum de *RFID* estão representadas na Figura 13 para a operação de forma adequada.

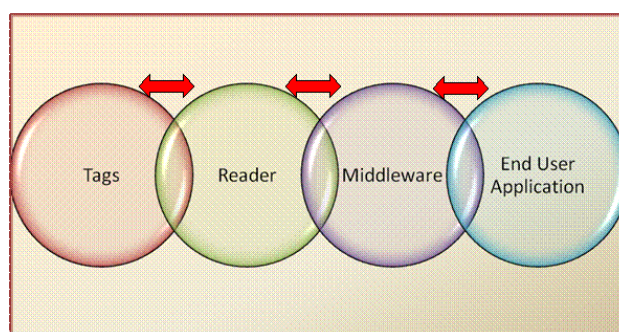


Figura 13 – Estrutura mínima de operação de um sistema de identificação *RFID*  
 Fonte: GLOVER AND BHATT (2006).

O *middleware* tem por objetivo tornar fácil a utilização dos dados obtidos através dos leitores *RFID*. Eles operam como gestores do fluxo destes dados, no processamento, na filtragem e na conectividade entre os *hardwares* (GLOVER; BHATT,2006).

Alguns aspectos devem ser levados em conta na hora de projetar um *middleware*:

- Suporte a múltiplos *hardwares*: O *middleware* deverá fornecer uma interface para conectividade entre diferentes tipos de *hardwares* e tecnologias.
- Sincronização: Deverá possuir uma sincronização entre todos os processos, pois desta forma minimizará a latência, melhorando sua eficiência.
- Manipulação em tempo real dos leitores *RFID*: O *middleware* também deve processar uma grande quantidade de dados capturados pelos leitores conectados em tempo real sem que haja erros de leitura.
- Interface com múltiplas aplicações: Deve ser capaz de interagir com aplicativos simultaneamente e atender todos requisitos.
- Escalabilidade: Deve permitir a integração de novos e diferentes dispositivos.
- Flexibilidade: Deve permitir a integração de novas e diferentes tecnologias (GLOVER; BHATT, 2006).

### 1.2.3.1. Componentes para o *middleware RFID*

Como o *middleware* é a interface padronizada entre diferentes *hardwares RFID* e as aplicações desenvolvidas para esta tecnologia de identificação eletrônica, ele deve fornecer algumas vantagens adicionais, como o processamento dos dados brutos antes de repassar estes dados tratados as aplicações, o fornecimento de uma interface para consulta dos dados e o

gerenciamento dos leitores, onde manipula as informações de *hardwares* sem precisar mostrá-las. Na Figura 14 pode-se observar uma interface de comunicação e padronização do sistema *RFID* de acordo com as especificações fornecidas pelo *middleware* (BURNELL,2008).

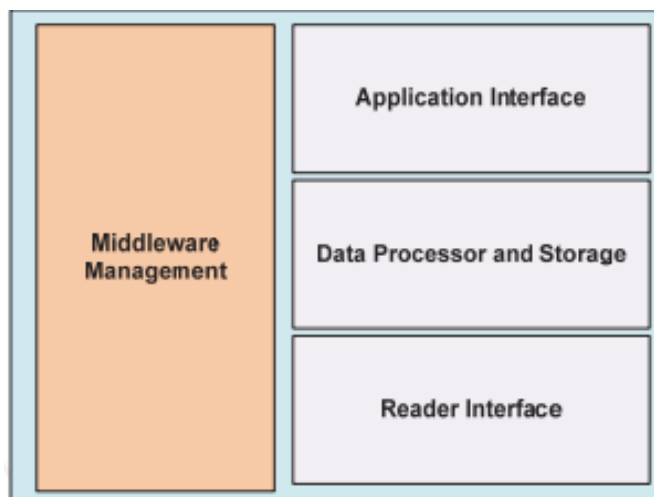


Figura 14 – Componentes típicos para um *middleware* de sistemas *RFID*  
 Fonte: BURNELL (2008).

Estes componentes típicos para a implementação de um *middleware* universal de sistemas de identificação eletrônica por radiofrequência podem ser divididos como:

- Interface do leitor (*Reader interface*): Mantém a conexão entre os diversos tipos, tecnologias e formas de leitores, mantendo padronizada a leitura de todos.
- Processamento de dados e armazenamento (*Data processor and storage*): É responsável pelo processamento e armazenamento dos dados brutos provenientes dos leitores.
- Interface de aplicação (*Application interface*): Recebe os dados já filtrados e os coloca no padrão estabelecido para enviar ao *software* e receber, suas informações como resposta de algum processamento.

### 1.3. Padrão *ZigBee*

O padrão *ZigBee* foi criado com objetivo de integrar vários dispositivos em uma rede de comunicação sem fio, utilizando a camada física (PHY) e de acesso (MAC) pelo padrão *IEEE 802.15.4*. As principais características da rede conhecida como *ZigBee* são o baixo consumo de energia, a operação nas frequências e faixas de 868 MHz e 20 kbps; 915 MHz e 40 kbps; 2,4 GHz e 250 kbps. Um mesmo nó de rede pode possuir diferentes atribuições.



### 1.3.1. Atribuição dos dispositivos *ZigBee*

Os dispositivos *ZigBee* podem receber diferentes atribuições na rede, dependendo da necessidade. Eles podem assumir as funções de coordenadores de rede, roteadores ou dispositivos finais. Coordenador (*Coordinator*): Dispositivo responsável por formar uma rede adaptativa, selecionar um canal de comunicação e um endereço específico. No coordenador é que ocorre a seleção de acesso aos roteadores e dispositivos finais. Roteador (*Router*): Oferece a rota de caminhos alternativos para transmitir os pacotes de dados. Dispositivo Final (*End Device*): Como característica tem baixo consumo de energia e está localizado no final da rede por não possuir funcionalidades adicionais (VASQUES, 2010). Na Figura 15 pode-se observar uma topologia de rede estrela implementada com a tecnologia *ZigBee*, onde os dispositivos coordenadores, roteadores e dispositivos finais estão interligados através de uma conexão sem fio. As topologias de rede nesta tecnologia podem ser estrela, árvore e malha.

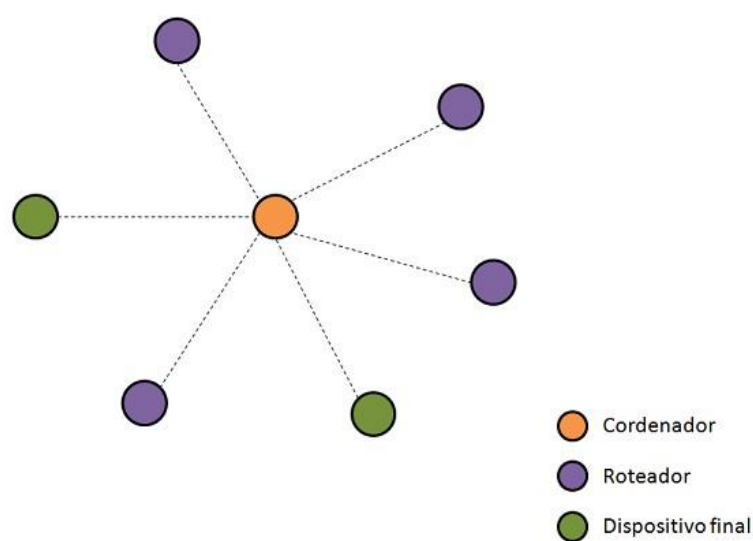


Figura 15 – Representação de uma rede típica WPAN de dispositivos *ZigBee*.  
Fonte: VASQUES (2015).

### 1.4. Topologia da rede padrão *ZigBee*

De acordo com as necessidades da aplicação define-se qual topologia de rede pode ser aplicada. Na topologia estrela a conexão é realizada entre os dispositivos e um único coordenador central. Na topologia malha conhecida também por *peer-to-peer* ou *mesh*, existe um único coordenador, mas todos os dispositivos podem comunicar-se entre si. Na topologia

árvore observa-se que o coordenador é o responsável pela rede, pois os dispositivos ligados ao coordenador são aglomerados de redes com a topologia malha.

### **1.5. Banco de dados MYSQL**

Segundo a Oracle, o MySQL é um sistema armazenamento de dados de dados relacional com a finalidade de organizar e preservar as informações. É a mais popular do mundo e possibilita a entrega de baixo custo, para aplicativos de banco de dados (ORACLE, 2015). O MYSQL é uma plataforma de código aberto utilizada na maioria das aplicações gratuitas. Ele utiliza a linguagem SQL (*Structure Query Language* – Linguagem de Consulta Estruturada), que segundo Pisa é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados (PISA, 2012).

### **1.6. Linguagem de programação PHP**

O termo *PHP* significa *Personal Home Page*, sendo definido como uma linguagem de alto nível. É uma linguagem de *script open source* de uso geral, utilizada mundialmente, e especialmente para o desenvolvimento *web*. Ela pode ser utilizada com a linguagem de marcação de texto o HTML (*Hypertext Markup Language*), que é executada através de um servidor (*http - Hyper- Text Transfer Protocol*). É uma linguagem gratuita, muito utilizada em ambientes *web* por possuir grande poder computacional, permitindo a construção de páginas dinâmicas. Uma característica essencial para o *PHP* é o fato de ele ser uma linguagem extremamente simples para um iniciante (PHPNET,2015).

## 2. MÉTODOS E TÉCNICAS

Embora este trabalho tenha sido originalmente idealizado para realizar a seleção de uma tecnologia de identificação eletrônica considerada ideal para aplicação em ambientes de clínicas e hospitais, foi observado que diferentes aplicações e requisições das demandas apresentam a necessidade utilização de diferentes tecnologias. Monitoramento e rastreamento de bens e processos em que os dispositivos de identificação (*tags*) não podem ser reutilizados ou possuem um grande volume, devem apresentar um baixo custo de aquisição e manutenção para viabilizar o sistema. Por outro lado, procedimentos ou monitoramento de operações de alto valor agregado devem ser mais elaborados e conter mais recursos, demandando um maior investimento de implementação ou manutenção.

Através desta nova visão sobre a utilização de modernas tecnologias de identificação eletrônica para a área da saúde, onde não se pode determinar apenas uma tecnologia específica para automação completa dos processos, buscou-se então integrar as diferentes técnicas de identificação para um único sistema de gerenciamento.

Como diferentes tecnologias precisam coexistir e operar de forma integrada para atender aos requisitos do projeto, verificou-se a necessidade de possuir uma plataforma universal de comunicações para a troca de dados. Nas pesquisas realizadas notou-se a falta de um *software* que fosse capaz de interagir com diferentes tecnologias para rastreabilidade. Desta forma, o objetivo do trabalho foi direcionado para o desenvolvimento de um *software (middleware)* para tratamento das informações recebidas dos leitores das três tecnologias selecionadas, sendo eles o *RFID*, o *ZigBee* e o código de barras. As pesquisas e experimentos realizados inicialmente sobre as tecnologias de identificação eletrônica foram então utilizados para complementar o desenvolvimento e os ensaios em laboratório do *software* de interface universal.

Como definição das técnicas de desenvolvimento para este trabalho dentro do cronograma proposto inicialmente, as atividades foram divididas em três diferentes etapas, destacando a pesquisa bibliográfica e desenvolvimento do protótipo do *middleware* de rastreabilidade de pacientes para tecnologias sem fio distintas, conforme a descrição mais detalhada neste capítulo.

Na primeira etapa de execução foi realizada a fundamentação teórica e prática das tecnologias aplicadas no *middleware*, a fundamentação teórica para desenvolvimento de um

*middleware* universal e a busca por soluções com equipamentos das tecnologias aplicadas neste trabalho (*RFID*, *ZigBee* e Código de Barras).

Com base nos resultados obtidos na etapa anterior de revisão bibliográfica, neste período foram definidos os recursos e ferramentas mais adequados para o desenvolvimento do *software* (*middleware*) e a realização dos ensaios em laboratório, assim como foram definidos os parâmetros relevantes das tecnologias para serem selecionadas pelo *middleware*.

Como última etapa de execução deste projeto, foi realizada a montagem dos equipamentos para a simulação de um ambiente real e o início dos ensaios de operação autônoma, coleta de dados de cada equipamento de tecnologia diferente, desenvolvimento e adequação do *middleware* a partir das coletas dos dados preliminares e a integração dos equipamentos com *software* e o ambiente de simulação. Adicionalmente para a otimização dos recursos disponíveis foi realizado o desenvolvimento de um simulador de coleta de dados da tecnologia *RFID*, com o objetivo de realizar os ensaios sem a presença dos equipamentos da Motorola. A partir dos dados adquiridos através das tecnologias utilizadas, desenvolveu-se o banco de dados para integração transparente de todos os sistemas de identificação e controle.

Concluindo este trabalho foram realizados os ensaios finais em laboratório para a análise do funcionamento do protótipo e eventuais correções de falhas. As informações obtidas foram utilizadas para a confecção deste documento final.

## **2.1. Ambiente de desenvolvimento**

O *middleware* para este projeto foi desenvolvido para tratar os dados coletados dos leitores com as tecnologias selecionadas como as mais indicadas para utilização em ambientes na área da saúde. Resumindo, este *middleware* tem como objetivo integrar as diferentes tecnologias de identificação eletrônica sem fio que podem ser utilizadas para ambientes hospitalares e clínicos. Os usuários são os pacientes que ao chegar ao local onde realizam os procedimentos clínicos, depois de confirmar seu cadastro, receberão algum tipo de identificador eletrônico, seja ele uma *tag RFID* em formato de pulseira, um código e barras em formato de pulseira ou então um cartão identificador onde encontra-se o dispositivo *ZigBee*. Em todos os locais onde são realizados os procedimentos e precisa ser confirmada a presença dos pacientes, haverá algum tipo de leitor para a tecnologia aplicada naquele local. Como pode-se utilizar as tecnologias *RFID*, *ZigBee* e leitor de código de barras para a automação dos procedimentos, o

*middleware* suporta estas três tecnologias, mas não está restrita somente a estas, pois foi desenvolvido para que suporte outras tecnologias existentes. A escolha da tecnologia a ser utilizada é livre, desde que atenda aos requisitos do *software*. O *middleware* foi desenvolvido levando em consideração os principais aspectos que ele deverá suportar.

### **2.1.1. Tecnologias e *hardwares***

Embora o objetivo principal deste projeto seja o desenvolvimento de um *software* de interface universal para aplicação em processos de identificação eletrônica de pacientes em ambientes de saúde, houve a necessidade de conhecer profundamente e operar todas as tecnologias utilizadas na automação. Por esse motivo, iniciaram-se as atividades com a configuração e operação de cada uma das tecnologias de identificação em separado. Neste capítulo estão sendo abordadas as configurações dos *hardwares* utilizados, como os leitores da tecnologia *RFID*, os leitores de códigos de barras de baixo custo e as redes adaptativas com tecnologia *ZigBee*. Com o domínio do sistema de comunicações e quadros de dados específicos de cada sistema que trafegam pelas redes sem fio, foram então abordadas também as ferramentas e tecnologias empregadas para o desenvolvimento do *middleware*, onde adicionalmente foram realizados ensaios com simuladores e *softwares* distribuídos comercialmente. Para a completa operação também foram desenvolvidos um banco de dados para o *middleware* e requisições, assim como um sistema tomador de decisões para a realização da coleta automática e filtragem de dados relevantes.

#### **2.1.1.1. Leitores de código de barras**

Inicialmente para a redução de custos do projeto, e tendo em vista que esta tecnologia de identificação eletrônica é amplamente conhecida e proliferada no varejo, buscou-se a utilização de aplicativos para utilização em dispositivos *mobile*. Para a realização das medições preliminares do sistema, foi utilizado um *smartphone Samsung Galaxy Note 2* como leitor, sendo que também foi utilizado nele um *software* para leitura de código de barras gratuito, instalado a partir da loja do *Googleplay*, fornecido pela *ZXing Team*, (ZXINGTEAM, 2015).

Algumas alterações para leitura do código de barras tiveram de ser realizadas no *software* proprietário. A primeira delas foi utilizar o IP da mesma rede onde ele se encontra

conectado, através da opção do campo pesquisa personalizada de URL. Desta forma configurou-se o caminho da rede, indicando para onde ele deverá enviar os dados adquiridos.

Como configurações gerais foram realizadas as seguintes modificações:

- IP configurado no *smartphone* como 10.0.42.10
- IP do servidor configurado como 10.0.42.1
- Configuração URL: `http://10.42.0.1/copatti/tcc_rfid/principal/recebecodebar/`

Na Figura 17 pode-se observar o aplicativo utilizado para a configuração do *software ZXing* no *smartphone*, redirecionando o envio dos dados para o local correto.

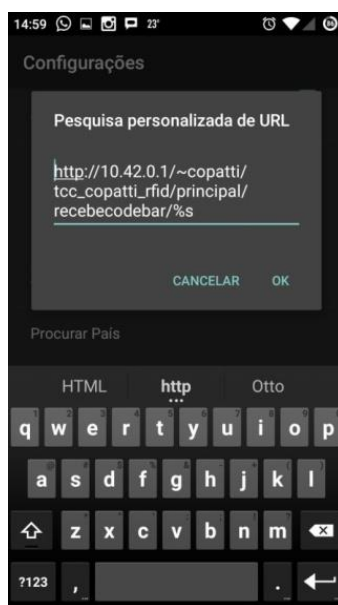


Figura 16 – Configuração do *software ZXing* para leitura do código de barras.

Fonte: Próprio Autor.

### 2.1.1.2. Leitores *RFID*

O leitor utilizado para os ensaios em laboratório foi o modelo Motorola-Symbol XR400. Utilizou-se este modelo por ser considerado um dos melhores equipamentos disponível no mercado. O Motorola-Symbol XR400 opera nas frequências UHF 902-928 MHz, e pode fazer leituras simultâneas de até quatro canais. Os protocolos compatíveis com este modelo de leitor *RFID* são o *EPCglobal*, Classe 0, Classe 0 (*read/write*), Classe 1 e Gen2 (SYMBOL/MOTOROLA, 2005). Na Tabela 6 pode-se verificar as características completas do leitor Motorola/Symbol XR400 utilizado no projeto.

Tabela 6 – Especificações do leitor *RFID* XR400 da Motorola.

CARACTERÍSTICAS	DESCRIÇÃO
<b>Características Operacionais</b>	
Frequência	Banda UHF, Frequências 902 ~928 MHz
Método	Frequency Hopping Spread Spectrum (FHSS)
Potência de Saída	30 dBm
Protocolo de Ligação Aérea	EPC Global, Classe 0, Classe 0 read.write, Classe 1 e Gen2
Sincronização	Network Time Protocol (NTP)
Endereçamento de IP	Estático e Dinâmico
<b>Conectividade</b>	
Rede	Conector RJ45 Ethernet 10/100T Conector Serial RS232
Energia	24 Vcc @ 1,2 A
Conector de RF	TNC Reverso
Pontos de Leitura (Antenas)	4 Antenas de Transmissão e 4 de Recepção
Dispositivos Externos	USB Primário USB Secundário Controle I/O Conector DB15
<b>Informações de Observação</b>	
Segurança	Certificação UL
Regulamentar	Região 1, FCC parte 15 US
<b>Hardware/OS e Firmware</b>	
Memória	64 MB de Flash e 64 MB de RAM
Sistema Operacional	Windows CE 4.2
Atualização de Firmware	Web
Protocolos de Gerenciamento	Suporte SNMP

Fonte: SYMBOL/MOTOROLA (2005)

Inicialmente para a operação do leitor *RFID* é necessária a conexão na rede, sendo os primeiros passos para configuração dos parâmetros como sendo os seguintes:

- Utiliza-se a porta *ethernet* para a troca de dados de identificação.
- Leitor *RFID* configurado como IP 10.0.0.1, com a máscara de rede 255.255.255.0.
- Configuração do IP no computador conectado ao leitor *RFID* como IP 10.0.0.2.
- Conector TX-1 destinado para a transmissão dos dados.
- Conector RX-1 destinado para a recepção.
- Conectar em uma fonte de alimentação de 24 Volts de corrente contínua.

Na Figura 16 pode-se observar o leitor *RFID* Motorola/Symbol XR400 e as conexões existentes do *hardware*, utilizadas para as antenas *RFID* e o servidor que contém o *middleware*. As antenas de interrogação e de leitura das informações são conectadas ao leitor em portas diferentes, permitindo a criação de diferentes configurações de áreas de cobertura de sinal, assim como orientações de posição ou de polarização, para a melhor seletividade dos *tags* que se deseja realizar a leitura automática de identificação.



Figura 17 – Ligações dos cabos para a operação do leitor *RFID* utilizado.  
Fonte: Próprio Autor.

### 2.1.1.3. Dispositivos *ZigBee*

Para esta tecnologia utilizaram-se os módulos fabricados pela *Digi International*, chamados de *XBee* modelo ZB serie 2. Estes módulos podem ter um alcance máximo de até 1.000 metros segundo o manual do *XBee*, embora sua distância de leitura possa ser configurada através da sua potência. Para as atribuições de coordenador, roteador e dispositivo final, o *software X-CTU* permite atualizar o *firmware* dos dispositivos em modo *API* ou *AT*. Através de outros comandos como PAN ID (*Personal Address Network*), cria-se uma rede privada, DH grava o endereço alto do destino, DL o endereço baixo do destino e NI a identificação do nó. Na Figura 18 pode-se observar as configurações do *ZigBee* através da utilização do *software X-CTU*, que é livremente distribuído pelo desenvolvedor da *Digi*.



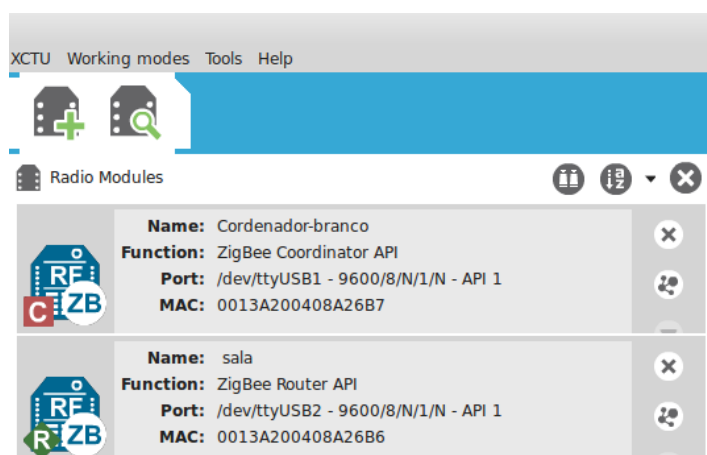


Figura 18 – Tela de configuração do *software* comercial distribuído pela Digi – X-CTU.  
Fonte: Próprio Autor.

## 2.2. Middleware

O *middleware* é um *software* que estabelece a conexão entre o leitor e o sistema de gerenciamento, onde faz a filtragem dos dados adquiridos através dos leitores. Deste modo entrega somente dados essenciais ao sistema de gerenciamento. Na pesquisa realizada para obter um *middleware* que suportasse diferentes tecnologias e que fosse escalável (suportar novas tecnologias existentes), encontrou-se um *middleware* que suporta somente uma tecnologia. Um exemplo é *Aker Middleware* da Topcode (TOPCODE, 2015), que suporta diferentes modelos de leitores *RFID* homologados, como está demonstrado na Figura 19.



Figura 19 – Página do desenvolvedor do *software* Aker Middleware - Compatibilidades  
Fonte: TOPCODE (2015).

Na Figura 20 pode-se observar o *software* disponível para a tecnologia *RFID* com os modelos de leitores utilizados neste projeto. Este *middleware* é um ótimo exemplo de aplicação que opera apenas com dados de uma tecnologia de identificação eletrônica, o que impossibilita a sua utilização para os requisitos deste projeto na área de saúde.

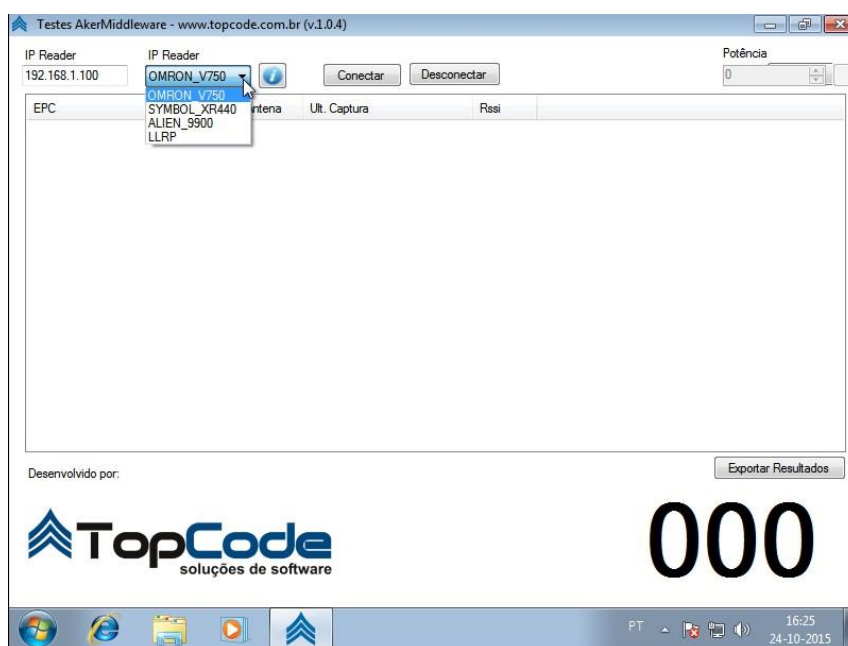


Figura 20 – *Software Aker Middleware RFID* com indicação do leitor XR440 da Motorola.  
Fonte: TOPCODE (2015).

O grande desafio deste desenvolvimento proposto é a criação de um *middleware* que suporte as diferentes tecnologias de comunicação sem fio, mas que também seja expansível e escalável, com conectividade para diferentes *hardwares*, levado em consideração:

- Suporte a múltiplos *hardwares*.
- Sincronização.
- Manipulação em tempo real do leitor.
- Interface com múltiplas aplicações.
- Escalabilidade.

Para modelar o banco de dados do *middleware* foram realizados alguns testes com os três tipos de leitores, e as informações obtidas destas tecnologias foram analisadas em um ambiente comum à todas elas. A Figura 21 representa os dados recebidos após os testes realizados para obter informações da leitora *RFID*.

```

<Metrics>
-<EventGroup>
-<TagList>
  <Tag raw="E2003412DC03011947234514" PC="3000" Reserved="0000000000000000" time="41d6649b" RPL="1"/>
</TagList>
</EventGroup>
</Metrics>

```

Figura 21 – Terminal com os dados recebidos através do sistema *RFID*.

Fonte: Próprio Autor.

A Figura 22 está demonstrando um exemplo das informações obtidas nos ensaios de laboratório realizados com a utilização do leitor de código de barras no *smartphone*.



Figura 22 – Aplicativo demonstrando os dados recebidos do código de barras.

Fonte: Próprio Autor.

Na Figura 23 pode-se observar a representação das informações coletadas e exibidas no terminal do servidor ao realizar o ensaio de leitura dos dados de identificação eletrônica com a tecnologia *ZigBee*, onde o endereço de identificação de 64 bits é apresentado.

```

root@Satellite: /home/copatti/public_html/tcc_copatti_rfid/application/models
File Edit View Search Terminal Help
root@Satellite:/home/copatti/public_html/tcc_copatti_rfid/application/models# python receive_samples.py
{'status': '\x00', 'frame_id': 'A', 'parameter': 'v\xcd\x00\x13\xa2\x00\x8a8\xb6 sala\x00\xff\xfe\x01\x00\xc1\x05\x10\xe1', 'command': 'ND', 'id': 't_response'}

```

Figura 23 – Terminal de visualização dos dados recebidos do sistema *ZigBee*.

Fonte: Próprio Autor.

A partir dos ensaios preliminares realizados com os leitores, foi identificado que alguns dados são comuns ao sistema externo e aos dados recebidos. O mais relevante de todos é o identificador de cada tecnologia, pois no *RFID* retorna o valor de uma *tag* em hexadecimal, já no código de barras retorna um valor decimal, e no *ZigBee* ele retorna seu endereço de 64 bits, que pode ser utilizado como identificador único. Neste caso as relações comuns entre o cadastro do exame e o cadastro no *middleware* são os identificadores de cada tecnologia. Outros campos comuns entre os dois sistemas são hora de entrada, hora de saída e sala de exames.

### 2.3. Banco de Dados

Embora a aquisição dos dados de forma padronizada para as diferentes tecnologias de identificação eletrônica utilizados represente um dos maiores desafios deste projeto, de forma semelhante também é necessário um cuidado especial ao arquivamento e tratamento das informações obtidas sobre os procedimentos monitorados. Receber estes dados e organizar os mesmos de forma que fiquem disponíveis para o tratamento e as tomadas de decisões de forma correta requer um amplo estudo, permitindo otimizar os espaços necessários e a rápida localização dos mesmos. Desta forma o projeto demanda a criação de um modelo de banco de dados para a realização dos ensaios com as tecnologias de identificação utilizadas.

O banco de dados foi modelado de acordo com as informações obtidas através do sistema de leitura das diferentes tecnologias, sendo que o relacionamento do sistema externo foi modelado com base nas informações correlacionadas com o *middleware* e as informações relevantes aos pacientes. A Figura 24 demonstra um diagrama que corresponde ao modelo ER do banco de dados do sistema externo que foi utilizado para realização dos ensaios em laboratório, onde pode-se visualizar as entradas dos dados dos pacientes.

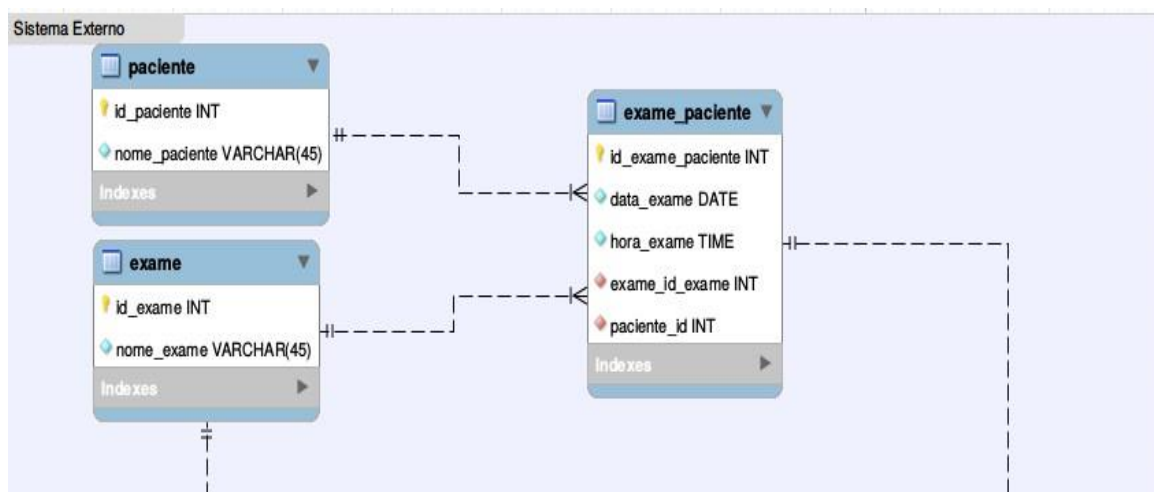


Figura 24 – Modelo ER do relacionamento externo do banco de dados utilizado.

Fonte: Próprio Autor.

As tabelas do banco de dados registram as informações obtidas através dos leitores que estão posicionados nas salas de exames, e a partir da aquisição das informações o *software* faz o tratamento e filtragem destes dados. Logo após filtrar os dados e obter somente os que são relevantes, ele realiza as inserções dos mesmos no banco de dados, em tabelas relacionadas.

A Figura 25 demonstra o modelo ER do sistema de *middleware* desenvolvido para este projeto, onde pode-se observar a presença das principais tabelas e relacionamentos que foram definidos de acordo com as requisições dos sistemas utilizados.

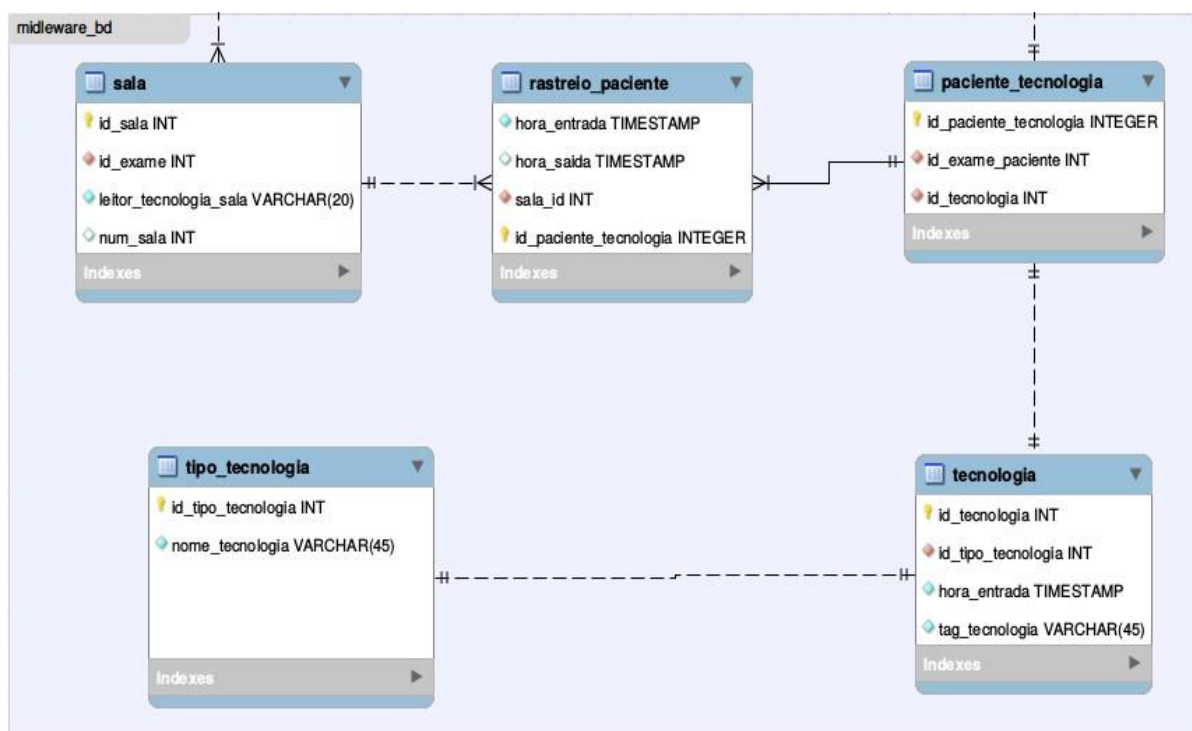


Figura 25 – Modelo ER do banco de dados utilizado no *middleware*.

Fonte: Próprio Autor.

De uma forma mais resumida, no modelo ER deste banco de dados pode-se verificar que na tabela “paciente\_tecnologia” tem-se uma relação de chave estrangeira com a tabela “exame\_paciente”. Esta relação está associada as informações de exames, o nome do paciente, a data do exame e a hora do exame. Também existe uma relação de chave estrangeira com a tabela “tecnologia”, onde são especificadas as informações das *tags* cadastradas no sistema.

Na tabela “tecnologia” existem os campos relacionados com a tabela “tag\_tecnologia”, que corresponde à identificação do paciente. Existe uma relação de chave estrangeira com a tabela “tipo\_tecnologia”, onde as opções nome da tecnologia que são utilizadas no sistema. Esta tabela “tipo\_tecnologia” realiza o registro dos tipos de tecnologias que são suportados no sistema, permitindo a flexibilidade das técnicas de identificação eletrônica que são suportadas. A tabela “sala” neste banco de dados possui uma relação de chave estrangeira entre as tabelas “paciente” e “exames”, onde a tabela “exames” faz conexão com o sistema externo, tendo como

campo principal o nome do exame. Na tabela “sala” encontram-se os campos “leitor\_tecnologia”, “número\_sala” e “id\_exame”, que é relacionada a uma sala para um exame específico somente. Neste caso, o leitor está vinculado a somente uma sala.

Concluindo a criação do banco de dados deste sistema encontra-se uma tabela chamada de “rastreo paciente”, onde são inseridos os dados da rastreabilidade do paciente, com o objetivo de verificar a localização imediata dos pacientes. As colunas informando “hora\_entrada” e “hora\_saída” possuem uma relação de chave estrangeira com a tabela “paciente\_tecnologia”, que se relaciona com as tabelas “tecnologia” e “exame\_tecnologia”.

## **2.4. Implementação do *software***

Para desenvolver a aplicação do *middleware* optou-se pela linguagem de programação *PHP* pelo motivo de ser uma linguagem de fácil entendimento, aprendizagem e ter boa documentação, uma vez que haveria pouco tempo para realizar todas as pesquisas bibliográficas das tecnologias envolvidas neste projeto, estudar estruturação do banco de dados, desenvolver a aplicação e fazer os ensaios. Da mesma forma optou-se pelo banco de dados *MYSQL*.

## **2.5. Ensaios para obtenção das informações dos leitores das tecnologias**

Após realizar os estudos sobre as tecnologias que devem ser suportadas no *middleware* desenvolvido, foram configurados todos os equipamentos de identificação eletrônica de acordo com as necessidades de cada aplicação para a realização dos ensaios em laboratório. Foram realizados os ensaios para a obtenção das informações transmitidas pelos leitores pela rede.

Os primeiros ensaios de laboratório com este sistema foram realizados com o leitor *RFID* da Motorola/Symbol XR400, obtendo as informações de leitura de *tags* UHF GEN2 passivas. Estas informações obtidas neste primeiro ensaio foram fundamentais para a base de construção do *middleware* flexível, de modo que suporte outros tipos de tecnologias de forma concomitante. Com o equipamento conectado ao servidor, foi executado o comando `curl http://10.0.0.1/cgi-bin/dataProxy?oper=queryTags` no terminal do *Linux*, onde este realiza a requisição dos dados ao leitor conectado na rede. Se uma *tag* estiver ao alcance da área de cobertura das antenas, este retornará os seguintes dados conforme demonstrado na Figura 26.

```

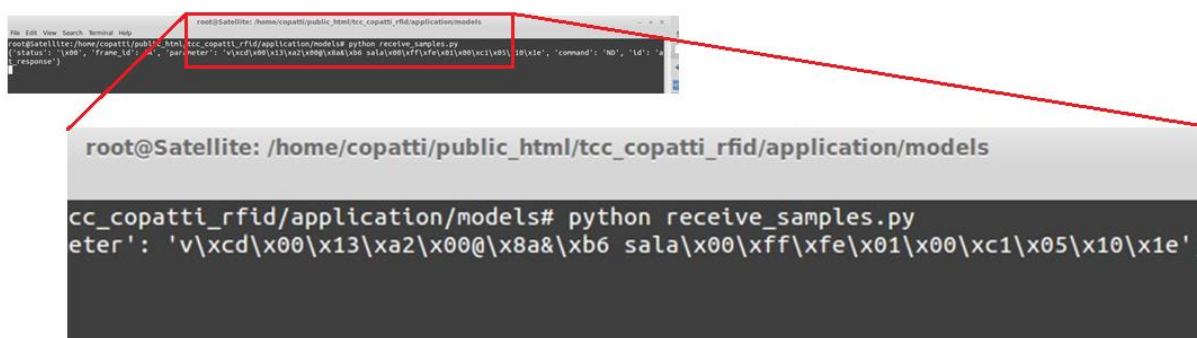
<Metrics>
-<EventGroup>
  -<TagList>
    <Tag raw="E2003412DC03011947234514" PC="3000" Reserved="0000000000000000" time="41d6649b" RPI="1"/>
  </TagList>
</EventGroup>
</Metrics>

```

Figura 26 – Terminal de leitura dos dados obtidos do *transponder RFID*.  
Fonte: Próprio Autor.

É importante observar que como todas as tecnologias deste projeto utilizam um identificador único, optou-se por utilizar este id como o principal elemento para filtrar os dados relevantes para o monitoramento e controle dos pacientes no ambiente dos ensaios.

Foi verificado que as *tags* utilizadas para credencial do sistema de aquisição dos leitores *RFID* retornaram como dados na rede de comunicação o seu identificador em hexadecimal, assim como o identificador do *ZigBee*, que também retorna seu endereço de rede de 64 bits em hexadecimal, mas com caracteres *ASCII* adicionais, como é possível visualizar na Figura 27. As leituras utilizando a tecnologia de código de barras retornaram apenas números decimais, pois foram utilizados como padrão os códigos de barras EAN-13.



```

root@Satellite: /home/copatti/public_html/tcc_copatti_rfid/application/models
cc_copatti_rfid/application/models# python receive_samples.py
[status: '\x00', 'frame_id': 'A', 'parameter': '\xcd\x00\x13\xa2\x00@\xa8\xb6 sala\xff\xfe\x01\x00\xc1\x05\x10\xe', 'command': 'ND', 'id': 'A', 'response']

```

```

root@Satellite: /home/copatti/public_html/tcc_copatti_rfid/application/models
cc_copatti_rfid/application/models# python receive_samples.py
eter': 'v\xcd\x00\x13\xa2\x00@\xa8\xb6 sala\xff\xfe\x01\x00\xc1\x05\x10\xe',

```

Figura 27 – Dados recebidos no terminal de identificação *ZigBee*.  
Fonte: Próprio Autor.

### 3. RESULTADOS E DISCUSSÃO

Considerando que o sistema e as tecnologias de identificação eletrônica já estavam devidamente desenvolvidos e testados em separado como forma de adquirir conhecimento sobre as características e limitações de cada um, foi iniciada a última etapa deste projeto, que corresponde a integração de todas as partes do sistema e a realização dos ensaios em laboratório com o sistema operando de forma autônoma. Com o tempo reduzido para realizar os ensaios e para minimizar a quantidade de variáveis para a análise dos resultados obtidos, optou-se por não realizar os ensaios finais do sistema em um ambiente real de utilização.

Para a aquisição dos dados dos leitores *RFID*, via TCP/IP, utilizou-se a porta *Ethernet* do leitor conectada ao cabo de par trançado com o servidor. Os dois dispositivos foram configurados na rede 10.0.0.0, sendo que o leitor da Motorola/Symbol XR400 foi configurado com IP 10.0.0.1 e o servidor com o IP 10.0.0.2. A Figura 28 demonstra o código desenvolvido para a aquisição das informações que são transmitidas pela rede na hora da leitura das *tags*.

```
// Le as informações do tranceiver.
$transURL = "http://10.0.0.1/cgi-bin/dataProxy?oper=queryTags";
#!/$transURL = "http://localhost/~copatti/tcc_copatti_rfid/TranceiverSimulator/refresh";

$xmlStr = file_get_contents($transURL);

// Interpreta como XML.
$xmlDoc = simplexml_load_string($xmlStr);

//echo "<pre>";

//print_r($xmlDoc->EventGroup->tagList->tag);

foreach ($xmlDoc->EventGroup->TagList->Tag as $tXml) {

    $suid=$tXml['raw'];
    $time=$tXml['time'];
    $antena=$tXml['RPL'];

?>
```

Figura 28 – Código para captura dos dados dos *tags RFID* utilizados.  
Fonte: Próprio Autor.

Para o leitor de código de barras foi utilizado um *smartphone* como *hardware* de leitura para fazer a aquisição e disponibilização dos dados de identificação, este estava conectado na mesma rede do servidor. Utilizou-se um *software* comercial para leitura dos códigos de barras, conforme pode-se observar na Figura 29 este código que faz a aquisição dos dados na rede.



```

<br><br><br>
<div class="container" >
<div class="row">
<div class="col-md-12">

<form action="<?php echo site_url('principal/recebecodebar');?>" method="post">
  <p>Código de barras:</p>
  (<a href="http://zxing.appspot.com/scan?ret=http://10.42.0.1/~copatti/tcc_copatti_rfid/principal/
  [codebar?codigo={CODE}]">Leitor</a>):
  <input type="text" name="cbcodebar" value="" />
</form>
<h1>assim</h1>
</div>
</div>
</div>

```

Figura 29 – Código utilizado para captura das informações do código de barras.

Fonte: Próprio Autor.

Por último foram realizadas as leituras com a tecnologia *ZigBee*, para obter as informações necessárias para que *middleware* fizesse a filtragem dos dados coletados. Utilizou-se uma *API* em *PHP* desenvolvida pela *Digi* em modo *API* do *ZigBee*. Devido a instabilidades desta *API*, não foi conseguida a estabilidade necessária para que esta enviasse o *frame* com os dados relevantes ao sistema, entre outras informações que retornavam. Para simplificar a execução destes ensaios através da conexão dos dispositivos, foi utilizado um código desenvolvido em *Python* (linguagem de programação de alto nível), apenas modificando alguns comandos básicos para obter as informações necessário ao sistema. A Figura 30 apresenta um fragmento do código em *Python* que foi utilizado, com as devidas alterações, para obter os dados relevantes para a operação do *middleware* com a tecnologia *ZigBee*.

```

from xbee import XBee
from serial import Serial
fh = open("teste2.txt", "w")
PORT = '/dev/ttyUSB1'
BAUD_RATE = 9600

# Open serial port
ser = Serial(PORT, BAUD_RATE)

# Create API object
xbee = XBee(ser)

xbee.at(frame_id='A', command='ND')
reply = xbee.wait_read_frame()
fh.write(str(reply))
fh.close()
print(reply)
# Getting the integer value out of reply
import struct
#print(struct.unpack('>h', reply['parameter'])[0])
# Continuously read and print packets
while True:
    try:
        response = xbee.wait_read_frame()
        print response
    "receive_samples.py" 45L, 875C

```

Figura 30 – Exemplo de parte do código para leitura do *transponder ZigBee*.

Fonte: Próprio Autor.

A resposta recebida no terminal do servidor para esta requisição de identificação está demonstrada na Figura 31, que descreve a comunicação da porta serial e a velocidade estabelecida. Ela também abre o arquivo de *log* que grava as informações transmitidas após o comando específico de solicitação de identificação "ND". A partir do dispositivo que está conectado na porta serial configurada, é enviada uma requisição para a rede, como se fosse uma pergunta “quem está aí?”, e a resposta é gravada no arquivo *log*. Este *log* é utilizado para tratar e obter somente os dados necessários para o sistema.

```
{'status': '\x00', 'frame_id': 'A', 'parameter': 'v\xcd\x00\x13\xa2\x00@\x8a&\xb6 sala\x00\xff\xfe\x01\x00\xc1\x05\x10\xe1', 'command': 'ND', 'id': 'at_response'}
```

Figura 31 – Terminal de demonstração dos dados obtidos da leitura do *ZigBee*.  
Fonte: Próprio Autor.

Após realizar o primeiro ensaio para obtenção das informações dos leitores, foi implementado um simulador utilizando as informações recebidas do leitor *RFID*. Ele utiliza o retorno do código xml como base para os dados do simulador. Como o mesmo código pode-se simular os dados também do código barras e do *ZigBee*. A Figura 32 apresenta parte do código do simulador utilizado, como forma mais rápida e simples de realizar os ensaios controlados.

```
<?php
class TranceiverSimulator extends CI_Controller {
public function refresh(){
echo '<?xml version="1.0"?>
<Metrics>
<!-- start-tag of Metrics XML document -->
<EventGroup>
<!-- start-tag of an event group -->
<ReadPointMap>
<!-- start tag for a Read Point Map -->
<pair name="Read Point #1" id="231" zone="First Door" class="Portal"/>
<pair name="Read Point #2" id="16"/>
<!-- running state determines the zone and class -->
<pair name="Read Point #3" id="790" zone="Second Door"/>
<!-- a new zone, running state determines the class -->
<pair name="Read Point #4" id="68"/>
<!-- running state determines the zone and class -->
</ReadPointMap>
<!-- end tag for a Read Point Map -->
<TagList>
<!-- start tag for purge list -->
<Tag raw="78" time="11111111" RPL="1"/>
<!-- running state -->
</TagList>
<!-- end tag for purge list -->
</EventGroup>
<!-- end-tag of an event group -->
</Metrics>
<!-- end-tag of Metrics XML document -->
';
}
}
```

Figura 32 – Código utilizado para executar o simulador da tecnologia *RFID*.  
Fonte: Próprio Autor.

No simulador utilizam-se as informações necessárias para representar a entrada e a saída de pacientes no ambiente clínico, com as informações da *tag*, tempo e antena de leitura. A Figura 33 apresenta um exemplo dos dados coletados que foram gerados pelo simulador.

The screenshot shows a web application interface titled 'Sistema de Rastreamento de Pacientes' with navigation links for 'Hospital', 'Exames', 'Raio-x', and 'Tomografia'. The main heading is 'TOMOGRAFIA'. Below it is a table with the following data:

Paciente	Exame	Data	Hora
João	Tomografia computadorizada	2015-12-03	02:05:00

Below this table is a section titled 'Paciente em atendimento' with another table:

Paciente	Tag	Exame	Hora Entrada	Hora Saída
João	E2003412DC03011947234514	Tomografia computadorizada	02:07:17	02:59:02

Figura 33 – Tela de demonstração do ensaio de rastreabilidade executado.  
Fonte: Próprio Autor.

A Figura 34 apresenta um exemplo de inserção dos dados na hora da leitura, que foram efetuados com sucesso e em tempo real, indicando o exame, a hora de entrada e a hora de saída que o paciente realizou o procedimento.

The screenshot shows a table interface with the following data and actions:

+ Opções		hora_entrada	hora_saida	sala_id	Id_paciente_tecnologia
<input type="checkbox"/>	<a href="#">Edita</a> <a href="#">Copiar</a> <a href="#">Apagar</a>	2015-12-03 03:08:20	2015-12-03 03:12:09	1	44
<input type="checkbox"/>	<a href="#">Edita</a> <a href="#">Copiar</a> <a href="#">Apagar</a>	2015-12-03 02:07:17	2015-12-03 02:59:02	1	39

Below the table are control elements: an up arrow, a 'Todos' checkbox, and the text 'Com os seleccionados:' followed by 'Muda' (with a pencil icon), 'Apagar' (with a red circle icon), and 'Exportar' (with a document icon).

Figura 34 – Tabela referente as informações de rastreio do paciente.  
Fonte: Próprio Autor.

Os ensaios foram realizados com o auxílio do simulador desenvolvido, que atua como um leitor *RFID*. Para poder inserir os dados automaticamente, um parâmetro de *refresh* foi inserido no código para atualizar a cada 120 segundos na página do Raio-x e Tomografia.

Quando estas páginas atualizam automaticamente neste período elas chamam uma função específica, que faz a inserção dos novos dados obtidos no banco.

O próximo ensaio que foi realizado necessitou de um *hardware* de leitura com o *smartphone*, que fez a função leitor de código de barras. Para a realização do ensaio foi cadastrado um paciente com um código de barras genérico, e posteriormente foi acessado o endereço para fazer a leitura do código de barras e enviar os dados processados ao sistema. A Figura 35 apresenta um exemplo da página com os resultados de leitura do código de barras.



Figura 35 – Tela do terminal de leitura do código de barras  
Fonte: Próprio Autor.

A Figura 36 apresenta os resultados obtidos com a hora de entrada e a hora de saída da sala de Raio-x, através da leitura da *tag* relacionada ao paciente e ao exame a ser realizado.

Sistema de Rastreamento de Pacientes				Hospital	Exames	Raio-x	Tomografia
Paciente	Exame	Data	Hora				
Nikita	Raio-x	2015-12-03	02:15:00				
Missio	Raio-x	2015-12-03	02:03:00				
Copatti	Raio-x	2015-12-03	15:01:00				
Marcio	Raio-x	2015-12-03	03:30:00				
Marcio	Raio-x	2015-12-03	03:30:00				
Paciente em atendimento							
Paciente	Tag	Exame	Hora Entrada	Hora Saída			
Copatti	001254E698AEC0078450121EF	Raio-x	03:08:20	03:22:58			
Marcio	4015000935708	Raio-x	03:29:42	03:41:24			

Figura 36 – Demonstração do *status* de entrada/saída do paciente com código de barras.  
Fonte: Próprio Autor.

Para a realização dos ensaios em laboratório com a tecnologia de identificação *RFID* utilizou-se o leitor *RFID* Motorola/Symbol XR400, conectado com duas antenas direcionais setorizadas, sendo uma TX de transmissão e uma RX de recepção. Estas antenas estão conectadas aos canais do dispositivo leitor. Foram utilizadas *tags* UHF passivas com tecnologia de padronização de segunda geração (*EPCglobal Gen2*). A identificação eletrônica das *tags* foram AAAAAAAAAABBBBBBBBBBCCCCCCCC e E2003412DC03011947234514, sendo que estes correspondem aos dados inseridos com sucesso. Neste momento notou-se que quando as *tags* não estavam sendo reconhecidas pelo leitor, retornava um erro no terminal. Este erro estava relacionado ao momento de leitura de uma *tag*, e como não existia uma *tag* próxima ao leitor, o mesmo retornava um campo vazio, verificando-se um erro no tratamento dos dados. Esse erro não causou em nenhum momento algum tipo de problema detectável para a leitura das *tags*. A Figura 37 apresenta o exemplo de uma destas leituras de informações de entrada e saída de pacientes em ambientes de exames, sendo realizadas com o equipamento *RFID* real da Motorola e não o simulador, utilizando as *tags* passivas do tipo *Gen2*.

The screenshot shows a web application interface for patient tracking. At the top, it says 'Sistema de Rastreamento de Pacientes' with navigation links for 'Hospital', 'Exames', 'Raio-x', and 'Tomografia'. The main heading is 'TOMOGRAFIA'. Below this is a table with columns: Paciente, Exame, Data, and Hora. The data row shows: João, Tomografia computadorizada, 2015-12-03, 11:29:00. Below this is another heading 'Paciente em atendimento' followed by a table with columns: Paciente, Tag, Exame, Hora Entrada, and Hora Saída. The data row shows: João, E2003412DC03011947234514, Tomografia computadorizada, 11:28:13, 11:28:36.

Paciente	Exame	Data	Hora
João	Tomografia computadorizada	2015-12-03	11:29:00

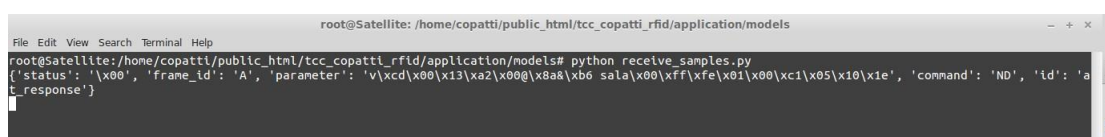
  

Paciente	Tag	Exame	Hora Entrada	Hora Saída
João	E2003412DC03011947234514	Tomografia computadorizada	11:28:13	11:28:36

Figura 37 – Tela do *middleware* com a leitura dos *transponders RFID* dos pacientes.  
Fonte: Próprio Autor.

Por último foram realizados os ensaios com os módulos *ZigBee* disponíveis para esta aplicação, sendo que um dos módulos estava configurado como coordenador e conectado na USB do servidor, e o outro *ZigBee* estava configurado como roteador e estava conectado em um computador portátil. A partir da execução do código de leitura de identificação dos nós da rede,

ele faz a requisição em *broadcast* para obter uma resposta se existe mais alguém na sala. Um *log* de eventos é gerado sempre que o coordenador receber as informações requisitadas a partir de um *frame* de dados, contendo as informações relevantes, como sua identificação através do endereço único de 64 bits. Na Figura 38 está sendo demonstrado um exemplo deste arquivo de *log* em formato txt, onde pode-se visualizar e analisar os dados relevantes contidos no *frame API* gerado automaticamente pelo módulo *ZigBee*.



```

root@Satellite: /home/copatti/public_html/tcc_copatti_rfid/application/models
File Edit View Search Terminal Help
root@Satellite:/home/copatti/public_html/tcc_copatti_rfid/application/models# python receive_samples.py
{'status': '\x00', 'frame_id': 'A', 'parameter': 'v\xcd\x00\x13\xa2\x00\x8a8\xb6 sala\x00\xff\xfe\x01\x05\x10\xe1', 'command': 'ND', 'id': 'a
t_response'}

```

Figura 38 – Arquivo de *log* demonstrando as informações relevantes do *frame API*

Fonte: Próprio Autor.

Na Figura 39 pode-se verificar a inserção dos dados no sistema que está identificando o paciente, a *tag*, a hora de entrada e a hora de saída. Pode-se verificar que o campo *parameter* do *log* apresenta um resultado igual ao do terminal após o retorno do comando de solicitação de descoberta de nós da rede. Este *frame* retornou seu endereço de 64 bits como “v xcd x00 x13 xa2 x00 x8a& xb6”, que pode ser utilizado como identificador para esta tecnologia.

Sistema de Rastreamento de Pacientes							
				Hospital	Exames	Raio-x	Tomografia
<b>TOMOGRAFIA</b>							
Paciente	Exame	Data	Hora				
João	Tomografia computadorizada	2015-12-03	11:29:00				
Juliana	Tomografia computadorizada	2015-12-03	12:05:00				
<b>Paciente em atendimento</b>							
Paciente	Tag	Exame	Hora Entrada	Hora Saída			
João	E2003412DC03011947234514	Tomografia computadorizada	11:28:13	11:28:37			
Juliana	0013a208b6	Tomografia computadorizada	12:05:00	12:14:49			

Figura 39 – Ensaios de operação do *middleware* realizados com os dispositivos *ZigBee*.

Fonte: Próprio Autor.

## CONCLUSÃO

Neste projeto foi demonstrado o desenvolvimento de um *middleware* de rastreabilidade de pacientes em ambientes clínicos e hospitalares, com suporte a diferentes tecnologias de redes sem fio. Baseando-se na premissa que este *middleware* deveria suportar a conexão com diferentes *hardwares* de identificação eletrônicas, ou seja, ser flexível para poder suportar outras tecnologias comerciais com os mesmos resultados, projetou-se um *middleware* universal para atender as tecnologias de radiofrequência - *RFID*, redes adaptativas no padrão *IEEE 805.15 - ZigBee* e o código de barras tradicional. Embora o projeto original tenha sido proposto para fazer uma comparação destas tecnologias e escolher qual destas se encaixaria para melhor automação de procedimentos e rastreabilidade em ambientes hospitalares, estes estudos serviram de base de conhecimento para o desenvolvimento do *software* de interface universal, assim como forneceu o suporte adequado para a realização dos ensaios em laboratório.

Como motivador deste desenvolvimento, durante a realização da pesquisa bibliográfica notou-se que existiam diversos *middlewares* com suporte para *hardwares* de uma mesma tecnologia apenas, desta forma não sendo compatíveis com a proposta deste projeto. Em casos da necessidade de uma implementação ambientes reais de utilização, uma empresa poderia optar por selecionar as soluções economicamente mais viáveis, mas estaria atrelado apenas aquela tecnologia específica e um *software* proprietário de gerenciamento. De um modo geral, ela deveria ter que realizar uma reestruturação da sua rede de comunicações, adquirindo novos produtos de *hardware*, *software* e infraestrutura da rede.

Para a implementação deste *middleware* universal foram enfrentados diversos desafios, que contaram como o suporte nos conhecimentos adquiridos durante o curso de Redes de Computadores e na elaboração das referências bibliográficas deste trabalho. Entre as dificuldades, por exemplo, foi realizar a conexão dos módulos com a tecnologia *ZigBee* para a aquisição das informações necessárias ao banco de dados, pois as *APIs* disponíveis em *PHP* que são disponibilizadas pelo fabricante não funcionaram de forma satisfatória, desta forma necessitando a utilização de outros métodos para poder realizar os ensaios com o *ZigBee*. Da mesma forma, um outro exemplo de dificuldade de desenvolvimento que foi encontrada estava relacionada com o modelamento e a criação do banco de dados, de forma que atendesse as necessidades e requisições do sistema proposto inicialmente.

Como resultados obtidos a partir dos ensaios realizados em laboratório, conclui-se que este *middleware* respondeu com desempenho satisfatório nas tarefas de integração das tecnologias de identificação eletrônica de pacientes propostas para este projeto.

Nos ensaios realizados para a requisição dos dados do leitor *RFID*, obteve-se de forma rápida a resposta entre o *middleware* e o leitor, sendo que este *middleware* filtrou as informações coletadas corretamente, somente apresentando os dados relevantes para que o sistema de gerenciamento conseguisse relacionar aos dados no banco de dados. Uma das grandes vantagens de se utilizar esta tecnologia associada ao *middleware* desenvolvido é que não necessita de visada direta com as antenas *RFID* para que as *tags* sejam lidas.

Da mesma forma, as requisições realizadas através do leitor de código de barras apresentaram os mesmos resultados, interagindo de forma satisfatória com *middleware*, enviando os dados coletados para a seleção correta das informações. Esta tecnologia permite disponibilizar ao sistema de gerenciamento somente as informações necessárias para relacionar com o banco de dados. A grande desvantagem desta tecnologia é a visada direta com o leitor.

Através da tecnologia de identificação com o *ZigBee* obteve-se também os resultados satisfatórios conforme proposto. Se comparada com as tecnologias *RFID* e código de barras, esta apresenta o maior tempo de resposta de leitura, pois não foi possível realizar a coleta das informações utilizando a linguagem *PHP* em que o *middleware* foi desenvolvido.

### **Sugestões para trabalhos futuros.**

Como este projeto possuía o objetivo de desenvolver um *middleware* para suportar as tecnologias de identificação distintas dentro de um período de tempo relativamente curto, estão sendo sugeridos algumas ações e aperfeiçoamento como sugestão de trabalhos futuros, baseados nas experiências e limitações que foram detectados durante a implementação deste trabalho:

- a) Desenvolver o *middleware* em uma linguagem mais adaptativa para o meio comercial.
- b) Pesquisar meios de segurança para aplicar no *middleware*, uma vez que muitas destas tecnologias são vulneráveis a ataques e há um grande fluxo de informações pessoais sendo transmitidas entre as tecnologias, o *middleware* e o restante do sistema.
- c) Desenvolver uma aplicação de gerenciamento de informações para gerar relatórios e estatísticas de desempenho e do fluxo de dados pela rede de comunicação.



## REFERÊNCIAS

ALMEIDA, L. C. **Aplicações da Tecnologia de Identificação por Rádio**. Trabalho Final de Curso (Engenharia de Teleinformática) - Universidade Federal do Ceará, Fortaleza, 2011.

BARCODING. **História do código de barras**. Disponível em:

<<http://www.barcoding.com/information/>>. Acesso em 20 de setembro de 2015.

BURNELL, J. **What Is RFID Middleware and Where Is It Needed?** Disponível em:

<<http://www.rfidupdate.com/articles/index.php?id=1176>>. Acesso em 15 de setembro de 2015.

DIGI. **XBee and XBee-PRO ZigBee RF Modules**. Disponível em:

<<http://www.digi.com/products/xbee-rf-solutions/modules/>>. Acesso em 10 de novembro de 2015.

FREITAS, D. R. R. **A importância de um middleware para o sistema RFID**. Disponível em: <<http://brasil.rfidjournal.com/artigos/vision?9940>>. Acesso em 11 novembro 2015.

GLOVER, B.; BHATT, H. **RFID Essentials**. [S.l.]: O'Reilly, 2006.

GLOVER, B.; BHATT, H. **Fundamentos de RFID**. Alta Books ed. Rio de Janeiro: Alta Books, 2008.

GOMES, H. M. C. **Construção de um sistema de RFID com fins de localização especiais**. 2007. Dissertação (Mestrado em Ciência da Computação) - Universidade de Aveiro, Aveiro Portugal, 2007.

GS1. **Types of EAN/UPC barcodes**. Disponível em: <<http://www.gs1.org/barcodes/ean-upc>>. Acesso em 29 de novembro de 2015.

GS1BR. **Chave GTIN**. Disponível em: <<https://www.gs1br.org/faq/o-que-e-gtin>>. Acesso em setembro de 2015.

GS1BRASIL. **GTIN - Número global do item comercial**. Disponível em:  
<<https://www.gs1br.org/codigos-e-padrees/chaves-de-identificacao/gtin>>. Acesso em  
1º de dezembro de 2015.

GS1BRASIL. **Chaves identificadoras**. Disponível em:  
<<https://www.gs1br.org/codigos-e-padrees/>>. Acesso em setembro de 2015.

IBM. **Controlando o RFID**. Disponível em: <<http://www.ibm.com/br/>>. Acesso em 29 de  
novembro de 2015.

IDESEG. **A história do RFID**. Disponível em:  
<<http://www2.idseg.com.br/home/porque-utilizar-rfid>>. Acesso em 20 de setembro de 2015.

INFO, G. **Global Trade Item Number**. Disponível em: <<http://www.gtin.info/gs1-us/>>.  
Acesso em 29 de novembro de 2015.

INSTRUMENTS, N. **Surge o novo padrão de RFID EPC Gen2**. Disponível em:  
<<http://www.ni.com/white-paper/13220/pt/>>. Acesso em 27 de novembro de 2015.

MARS, R. **A história do código de barras**. Disponível em:  
<<http://gizmodo.uol.com.br/a-historia-nao-contada-da-origem-dos-codigos-de-barras/>>.  
Acesso em 29 de setembro de 2015.

OLIVEIRA, S. **Código de barras - Pura matemática!** Disponível em:  
<<https://oprofessorweb.wordpress.com>>. Acesso em 17 de outubro de 2015.

ORACLE. **O Banco de Dados de Código Aberto Mais Popular do Mundo**.  
Disponível em: <<http://www.oracle.com/br/products/mysql/overview/index.html>>.  
Acesso em 29 de outubro de 2015.

PHILLIPS, T. K. B. E. G. B. L. B. T. **Guidelines for Securing Radio Frequency  
Identification (RFID) Systems**, 2007.

PHPNET. **O que é PHP?** Disponível em: <<https://www.php.net/>>. Acesso em 29 de outubro de 2015.

PISA, P. **O que é e como usar o MYSQL?** Disponível em: <<http://www.techtudo.com.br/artigos/noticia/2012/04/>>. Acesso em 29 de outubro de 2015.

RFIDPOINT. **El estándar EPC.** Disponível em: <<http://www.rfidpoint.com/fundamentos/el-estandar-epc/>>. Acesso em 30 outubro 2015.

SANTINI, A. G. **RFID Conceitos, aplicabilidades e impactos 1.ed.** 1.ed. Rio de Janeiro: Rio de Janeiro: Ciência Moderna, 2008. (Ciência Moderna).

SANTOS P. J. M dos. **Identificação por Radiofrequência: aplicações e vulnerabilidades da tecnologia RFID.** Disponível em: <<http://web.unifoa.edu.br/cadernos/edicao/02/18.pdf>>. Acesso em 29 de novembro de 2015. Sequencial de Redes de Computadores - UniFOA.

SWEDBERG, C. **Hotel suíço oferece serviço cinco estrelas com tecnologia.** Disponível em: <<http://brasil.rfidjournal.com/noticias/vision?11115>>. Acesso em 15 de outubro de 2015.

SYMBOL/MOTOROLA. **XR400 RFID Reader Integrator Guide.** 2005.

TOPCODE. **Aker Middleware RFID.** Disponível em: <<http://www.topcode.com.br/wp/index.php/aker-middleware-rfid/>>. Acesso em 02 de dezembro de 2015.

VASQUES, Bruna Luísa Ramos P. **Topologia ZigBee.** Disponível em: <[http://www.gta.ufrj.br/grad/10\\_1/zigbee/](http://www.gta.ufrj.br/grad/10_1/zigbee/)>. Acesso em 29 de novembro de 2015.

ZXINGTEAM. **Barcode Scanner.** Disponível em: <<https://play.google.com/store>>. Acesso em 29 de novembro de 2015.