

Universidade Federal do Maranhão  
Centro de Ciências Exatas e Tecnológicas  
Curso de Ciência da Computação

**JOSÉ NUNES DE OLIVEIRA NETO**

**SISTEMA COLABORATIVO DE COTAÇÃO E  
COMPARAÇÃO DE PREÇOS UTILIZANDO NEAR  
FIELD COMMUNICATION**

São Luís  
2014

**JOSÉ NUNES DE OLIVEIRA NETO**

**SISTEMA COLABORATIVO DE COTAÇÃO E  
COMPARAÇÃO DE PREÇOS UTILIZANDO NEAR  
FIELD COMMUNICATION**

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof<sup>o</sup> Marcelo Vidigal Gonçalves dos Reis

São Luís

2014

Oliveira Neto, José Nunes de

Sistema Colaborativo de Cotação e Comparação de Preços Utilizando Near Field Communication/ José Nunes de Oliveira Neto. – 2014.

70 p.

Impresso por computador (Fotocópia).

Orientador: Marcelo Vidigal Gonçalves dos Reis

Monografia (Graduação) – Universidade Federal do Maranhão, Curso de Ciência da Computação, 2014.

1. NFC 2. Dispositivos móveis 3. Sistemas colaborativos 4. Cotação 5. Android I.  
Sistema Colaborativo de Cotação e Comparação de Preços Utilizando Near Field Communication

CDU 004.42

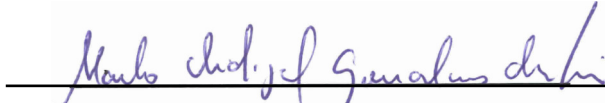
**JOSÉ NUNES DE OLIVEIRA NETO**

**SISTEMA COLABORATIVO DE COTAÇÃO E COMPARAÇÃO  
DE PREÇOS UTILIZANDO NEAR FIELD COMMUNICATION**

Monografia apresentada ao curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.


Aprovado em 26 de Junho de 2014.

BANCA EXAMINADORA



**Profº Marcelo Vidigal Gonçalves dos  
Reis (Orientador)**

Instituto Federal Educação, Ciência e  
Tecnologia do Maranhão



**Profª Simara Vieira da Rocha**  
Universidade Federal do Maranhão



**Profº Bruno Feres de Souza**  
Universidade Federal do Maranhão

*Aos meus pais e irmãos,  
que são meu esteio...*

# Agradecimentos

Agradeço primeiramente a Deus por ter me dado força em cada dia da minha vida para que eu pudesse caminhar e estudar, assim chegando hoje onde estou.

Agradeço aos meus avós. À minha avó Itaci, por ter cuidado de mim desde pequeno e me ensinado tantas coisas interessantes sobre como era viver no tempo dela. À minha avó Elizabeth, por ter sempre sorrido ao me ver e por sempre me tratar tão bem. Em especial ao meu avô José Nunes, que infelizmente já não se encontra mais aqui entre nós, mas que sempre torceu por mim e sempre me aconselhou coisas boas sobre como ser um homem digno e de quem pude herdar também o mesmo nome.

Agradeço aos meus pais, por terem me criado e me dado educação durante todos esses anos para que eu pudesse ser quem sou hoje. Ao meu pai, que sempre foi aquele cara brincalhão, que mais parecia um irmão, mas que sempre confiou em mim. À minha mãe, que é uma mulher sábia e a quem me inspirei em boa parte dos pensamentos que possuo.

Agradeço aos meus irmãos, pois suas brincadeiras puderam fazer esquecer dos meus problemas e suas brigas me puxaram as orelhas para que eu tivesse cada dia mais vontade de me graduar.

Agradeço aos meus amigos, pois sem amigos para torcer por mim, para me fazer ter força, eu também não conseguiria. Aos amigos da época de cursinho, por eles me ajudaram a entrar na faculdade. Aos amigos de faculdade, pois eles me ajudaram a ter êxito nas disciplinas. Aos meus amigos de trabalho, pelos cafés e ajuda na monografia.

Agradeço ao meu orientador, Marcelo Vidigal, pelas conversas, orientações e pelo tema proposto.

Agradeço aos professores da Universidade Federal do Maranhão e do Instituto Federal de Educação, Ciência e Tecnologia do Maranhão, pois sem eles, não estaria aqui sentindo-me mais confortável em redigir assuntos e temas da área de Computação.

À Eveline Sá, Jeane Teixeira e Mauro Silva, por ter tido oportunidade de conversar com eles antes da definição da minha monografia. À Josenildo Silva, Santiago Sinézio e Karla Fook, por terem me ajudado na estrutura e escrita da monografia, tirando eventuais dúvidas.

Aos alunos do Instituto Federal de Educação, Ciência e Tecnologia do Maranhão, em especial os alunos do ensino técnico de informática, pois a cada dia que conheço um pouco mais da vida de cada um, me inspiro e tenho forças para seguir em frente.

*“Não vos amoldeis às estruturas deste mundo,  
mas transformai-vos pela renovação da mente,  
a fim de distinguir qual é a vontade de Deus:  
o que é bom, o que Lhe é agradável, o que é perfeito.  
(Bíblia Sagrada, Romanos 12, 2)*

# RESUMO

Grande parte das pessoas, antes de adquirir um produto, tende a pesquisar e comparar preços. Isso ocorre devido a diversos fatores como, por exemplo, a necessidade de economizar capital. Diversos sistemas computacionais existem no mercado para auxiliar essas pessoas. Em geral, essas soluções atendem em parte as necessidades dessas pessoas, pois em sua maioria necessitam de conexão à internet e poucas exploram a colaboração. A colaboração torna possível que os resultados sejam compartilhados entre os participantes, com pouco esforço para a realização da tarefa. Desta forma, sistemas colaborativos existem para auxiliar na execução de objetivos e tarefas de um determinado problema, utilizando ferramentas para obtenção de soluções, como computadores e dispositivos móveis. Um exemplo de dispositivo móvel são os *smartphones*, que a cada vez mais estão inseridos no cotidiano das pessoas. No Brasil, existe uma elevada quantidade de *smartphones*. *Smartphones* e *tablets* são dispositivos com poder computacional em tamanho reduzido que possuem diversas tecnologias agregadas. Muitas dessas tecnologias, como Wi-Fi<sup>TM</sup> e Bluetooth<sup>®</sup>, favorecem tarefas como o compartilhamento de informações e o acesso à internet. Novas tecnologias, como o NFC (*Near Field Communication*), vem também surgindo para facilitar essas tarefas. NFC é uma tecnologia inovadora com boa proposta, através de uma intuitiva forma de interação. Basta aproximar dois dispositivos para que uma comunicação se realize entre eles. Este trabalho propõe uma solução colaborativa para o problema da cotação e comparação de preços utilizando dispositivos móveis e a tecnologia NFC. Além disso, este trabalho se propõe a apresentar com detalhes a tecnologia NFC, expondo sua arquitetura bem como apresentando aspectos em que aplicações colaborativas beneficiam-se do uso do NFC.

**Palavras-chaves:** NFC. Dispositivos móveis. Sistemas colaborativos. Cotação. Android.



# ABSTRACT

Most people search and compare prices before purchasing a product. This is due to several factors such as the need to save capital. There are several computational systems to help these people. In general, these solutions resolve in part the problem, because most of them require an internet connection and few explore collaboration. The collaboration makes possible to share the results among participants, with little effort to perform the task. Thus, collaborative systems are to assist in the implementation of goals and tasks of a problem characterized using tools to obtain solutions, such as computers and mobile devices. An example of mobile are the smartphones that are increasingly embedded in daily life. In Brazil, there is a high amount of smartphones. Smartphones and tablets are devices with computational power in a smaller size that have aggregated several technologies. Many of these technologies, such as Wi-Fi<sup>™</sup> and Bluetooth<sup>®</sup> favor tasks such as sharing information and Internet access. New technologies such as NFC (from Near Field Communication) is also emerging to facilitate these tasks. NFC is an innovative technology with good proposal and an intuitive way of interaction. Just approach two devices to have a communication between them. This paper proposes a collaborative solution to the problem of price quoting and comparison using mobile devices and NFC technology. Furthermore, this paper proposes to present in detail the NFC, exposing its architecture as well as presenting ways in which collaborative applications benefit from the use of NFC.

**Keywords:** NFC. Mobile. Collaboration. Citation. Android.

# Lista de ilustrações

Figura 1 – Representação do Modelo 3C. . . . .	21
Figura 2 – <i>Transponders</i> Ativos e Passivos - Sistema RFID . . . . .	27
Figura 3 – Princípio de Funcionamento do NFC: Acoplamento Indutivo. . . . .	29
Figura 4 – Representação do suporte do NFC à quatro tipos de <i>tags</i> . . . . .	30
Figura 5 – Representação simplificada do formato NDEF. . . . .	32
Figura 6 – Estrutura do primeiro octeto de um registro NDEF. . . . .	33
Figura 7 – NFC e suas diversas aplicações. . . . .	41
Figura 8 – Ciclo de Vida de uma Activity no Android™ . . . . .	47
Figura 9 – Cenário Reduzido onde o Priceshare foi idealizado. . . . .	51
Figura 10 – Arquitetura do <i>software</i> Priceshare. . . . .	52
Figura 11 – Modelo conceitual dos dados da aplicação Priceshare. . . . .	53
Figura 12 – Modelo lógico gerado a partir do modelo conceitual do Priceshare. . . . .	54
Figura 13 – Diagrama de Caso de Uso para um usuário do Priceshare. . . . .	56
Figura 14 – Diagrama de Sequência para ilustrar o caso de uso “Trocar Lista de Desejo com outro usuário”. . . . .	57
Figura 15 – Diagrama de Classes do Priceshare. . . . .	57
Figura 16 – Tela de cadastro de produtos. . . . .	61
Figura 17 – Exibição de produtos de uma lista. . . . .	61
Figura 18 – Fluxograma representando a importação de uma lista recebida de outro usuário. . . . .	62
Figura 19 – Ambiente utilizado para testes usando OpenNFC™ e VirtualBox. . . . .	62
Figura 20 – Análise entre Tamanho dos dados e Tempo durante uma transmissão em Modo <i>Peer-to-peer</i> . . . . .	64

# Lista de códigos

Código 1 – Modelo Físico em linguagem SQL do esquema para SQLite. . . . .	54
Código 2 – Trecho da classe Produto com anotações Java para uso com ORMLite	59
Código 3 – Trecho de código para inicialização do NFC usando Android Beam. . .	60

# Lista de tabelas

Tabela 1 – Valores para o campo TNF. . . . .	34
Tabela 2 – Tabela de Comparação entre Tecnologias de Rede. . . . .	34
Tabela 3 – Afinidade de Tipos de Dados para o SQLite . . . . .	55

# Lista de abreviaturas e siglas

ADT	Android Development Tools
API	Application Programming Interface
APK	Android Package File
AVD	Android Virtual Device
CF	Chunk Flag
CNP	Cadastro Nacional de Produtos
CRUD	Create, Retrieve, Update and Delete
CSCW	Computer Supported Cooperative Work
DAO	Data Access Object
DCF	Document Computation Format
DEX	Dalvik Executable
EDR	Enhanced Data Rate
HTML	HyperText Markup Language
IDE	Integrated Development Environment
IEC	International Electrotechnical Commission
IFF	Identify Friend or Foe
IL	Identification Length
IrDA	Infrared Data Association
ISO	International Organization for Standardization
MB	Message Begin
ME	Message End
MIME	Multipurpose Internet Mail Extensions
MVC	Model-view-controller

NFC	Near Field Communication
NDEF	NFC Data Exchange Format
OHA	Open Handset Alliance
ORM	Object-relational mapping
PAN	Personal Area Network
P2P	Peer-to-peer
PDA	Personal Digital Assistant
PDF	Portable Document File
RF	Radio Frequency
RFID	Radio Frequency Identification
SDK	Software Development Kit
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	Structured Query Language
SR	Short Record
TNF	Type Name Format
UML	Unified Modeling Language
URI	Uniform Resource Identifier
WLAN	Wireless Local Area Network
WYSIWYG	What You See Is What You Get
XLS	Microsoft Excel Spreadsheet Extension
XML	eXtensible Markup Language

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>19</b>
<b>2.1</b>	<b>Sistemas colaborativos</b>	<b>19</b>
2.1.1	Introdução	19
2.1.2	Groupware	19
2.1.3	Modelo 3C	20
<b>2.2</b>	<b>Sistemas de cotação de preços</b>	<b>23</b>
<b>3</b>	<b>NEAR FIELD COMMUNICATION</b>	<b>26</b>
<b>3.1</b>	<b>Contextualização</b>	<b>26</b>
<b>3.2</b>	<b>NFC forum</b>	<b>27</b>
<b>3.3</b>	<b>Aspectos técnicos</b>	<b>28</b>
<b>3.4</b>	<b>Comparação com outras tecnologias</b>	<b>33</b>
3.4.1	Bluetooth	34
3.4.2	Wi-Fi	35
3.4.3	ZigBee	36
3.4.4	iRDA	37
<b>3.5</b>	<b>Segurança</b>	<b>37</b>
<b>3.6</b>	<b>Aplicações</b>	<b>40</b>
<b>4</b>	<b>A PLATAFORMA ANDROID</b>	<b>43</b>
<b>4.1</b>	<b>Introdução</b>	<b>43</b>
<b>4.2</b>	<b>Componentes e tecnologias</b>	<b>45</b>
4.2.1	View	45
4.2.2	Activity	45
4.2.3	Intent	46
4.2.4	Handler	46
4.2.5	NFC e Android	48
<b>5</b>	<b>PRICESHARE</b>	<b>50</b>
<b>5.1</b>	<b>Arquitetura do sistema</b>	<b>50</b>
<b>5.2</b>	<b>Modelagem do software</b>	<b>52</b>
<b>5.3</b>	<b>Aspectos de implementação do software</b>	<b>58</b>
<b>5.4</b>	<b>Testes e resultados</b>	<b>61</b>
<b>6</b>	<b>CONCLUSÃO</b>	<b>65</b>

<b>Referências . . . . .</b>	<b>67</b>
------------------------------	-----------



# 1 Introdução

Uma grande parcela da população tem o hábito de pesquisar e comparar preços antes de adquirir o produto desejado. Fatores como inflação, condição econômica ou estilo de vida fazem com que essas pessoas queiram economizar.

Porém, pesquisar o preço de produtos pode ser uma tarefa difícil. A distância entre lojas, vendedores mal treinados, carência de produtos ou falta de tempo para pesquisar são alguns dos muitos problemas que as pessoas podem encontrar ao cotar preços.

Empresas também pesquisam preços dos produtos antes de adquiri-los. Existem diversos *softwares* que fornecem cotações e comparações de preços dos mais variados produtos às empresas. Este tipo de *software* tem por objetivo reunir informações de produtos e/ou serviços oferecidos por diversas empresas e exibir aos interessados essas informações de forma que o usuário consiga visualizar a melhor oferta, quanto a critérios como preço e qualidade. São exemplos de *softwares* para cotação e comparação de preços as aplicações Rede Cotação (Dicto Sistemas, 2013), Custo Certo (Custo Certo, 2013) e PriMus-DCF (Acca Software, 2014). Em sua maioria, são *softwares* sob licenças pagas, mas podemos encontrar opções gratuitas como o PriMus-DCF. O PriMus-DCF possui uma interface simples e possui um editor WYSIWYG (do acrônimo em inglês *What You See Is What You Get*), que serve para criação do modelo de visualização dos dados. Além disso possui um formato próprio para representação de seus dados e compartilhamento de listas de orçamentos e preços, denominado DCF (*Document Computation Format*). Os usuários do *software* podem também exportar em diversos outros formatos, como planilhas do Microsoft Excel (formato XLS), documentos de visualização portátil (formato PDF) e páginas web (formato HTML).

Em geral, os *softwares* já citados não atingem o usuário comum, pois foram projetados com diversas funcionalidades visando empresas e compras em grande volume. Assim, as pessoas tendem a continuar com os métodos tradicionais de pesquisa de preço como, por exemplo, anotações, navegação em *sites* de comércio eletrônico e ligações telefônicas.

Em se tratando de comércio eletrônico, há sistemas específicos para comparação de preços nesse tipo de loja, como Buscapé (BUSCAPÉ, 2014). O Buscapé é um *software* que pesquisa os preços dos produtos em diversos *sites* registrados em sua base de dados e retorna ao usuário a sugestão de menor preço. Com isso, algumas pessoas realizam uma busca prévia pelo preço do produto desejado antes de ir a uma loja física comprar o produto. Com o conhecimento do menor preço, as pessoas podem utilizar esse valor como base para realizar a compra. É importante salientar que o Buscapé apenas se baseia em

preços de *sites* de comércio eletrônico e não trabalha de forma colaborativa, uma vez que se trata de um sistema inteligente que opera de forma computadorizada.

Lojas físicas de algumas regiões do Brasil e algumas praças do setor comercial ainda não utilizam sistemas de comércio eletrônico. Alguns dos fatores que ainda distanciam essas lojas do comércio eletrônico são a falta de conhecimento desta prática ou o próprio estilo de negócio. Assim, as pessoas que desejam economizar se deslocam de loja em loja pesquisando os preços e anotando, por exemplo, em um papel ou no *smartphone*. No fim, pode analisar-se qual loja ofereceu o produto pelo menor preço e realizar a compra. A cada dia, inúmeras pessoas realizam este mesmo procedimento. É possível perceber que se essas informações pudessem ser compartilhadas, conseguiríamos agilidade, comodidade e menos esforço. As pessoas poderiam compartilhar suas cotações de preços e receber a sugestão de menor preço do produto desejado utilizando, como exemplo, um sistema rodando em seus *smartphones*.

*Smartphones* estão cada vez mais inseridos no cotidiano das pessoas. O Brasil é o quarto país do mundo em número de smartphones, aproximadamente 70 milhões (MEEKER, 2014). *Smartphones* e *tablets* são computadores, de tamanho reduzido, com um variado número de tecnologias acoplado a eles. A cada nova versão de *smartphones* e *tablets*, novas tecnologias e melhorias surgem. Muitas dessas tecnologias, como Wi-Fi<sup>TM</sup> e Bluetooth<sup>®</sup>, favorecem o compartilhamento de informações e o acesso à rede mundial de computadores.

Dentre as novas tecnologias, temos o NFC, do inglês *Near Field Communication*. NFC é uma tecnologia inovadora, com uma idéia simples. Basta aproximar dois dispositivos para que uma comunicação se realize entre eles. Assim, trocar arquivos e informações se torna prático e seguro, uma vez que a comunicação só é realizada em curta distância entre os dispositivos.

Neste contexto, este trabalho pretende investigar uma solução para o problema de compartilhamento de cotações de preços e comparação das informações trocadas entre usuários, por meio da tecnologia NFC presente em *smartphones* e *tablets*.

O objetivo geral deste trabalho é:

- a) Apresentar uma aplicação colaborativa para cotação e comparação de preços utilizando a tecnologia *Near Field Communication*.

Em seguida, enumera-se os objetivos específicos deste trabalho:

- a) Analisar a tecnologia *Near Field Communication*, identificando prós, contras e áreas de aplicação;

- b) Compreender o funcionamento e API da tecnologia *Near Field Communication* no sistema operacional Android<sup>TM</sup> para desenvolvimento de aplicativos;
- c) Verificar o uso da tecnologia *Near Field Communication* em sistemas colaborativos.

Este trabalho organiza-se de forma a dar a melhor compreensão do assunto abordado. No primeiro capítulo, é exposto uma breve introdução do problema, sua motivação, objetivo geral e objetivos específicos. No segundo capítulo, temos a fundamentação teórica deste trabalho, com os assuntos pertinentes a caracterização e solução do problema, como Sistemas de Cotação de Preços e Sistemas Colaborativos. No terceiro capítulo, apresentamos um estudo sobre o NFC, abordando aspectos técnicos e de segurança, comparações com outras tecnologias e seu atual uso em aplicações. No quarto capítulo, estudamos o sistema operacional Android<sup>TM</sup>, apresentando a arquitetura e os principais componentes utilizados para o desenvolvimento da solução proposta neste trabalho. No quinto capítulo, veremos o *Priceshare*, uma aplicação colaborativa para o problema de cotação e comparação de preços, sua arquitetura, modelos e aspectos de implementação da aplicação. No sexto capítulo, temos a conclusão para este trabalho e relação de trabalhos futuros.

## 2 Fundamentação Teórica

### 2.1 Sistemas Colaborativos

Nesta seção entenderemos melhor o que é um Sistema Colaborativo, e quais as características necessárias para a classificação de um *software* como sistema colaborativo.

#### 2.1.1 Introdução

Com a sociedade cada vez mais conectada através do uso da *internet*, os *softwares* deixam de serem meros processadores de dados para serem verdadeiros sistemas de colaboração. Isso se dá porque a maior parte dos problemas são complexos, surgindo a necessidade de termos uma quantidade de pessoas alocadas para trabalhar em prol da solução de um problema. (PIMENTEL; FUKS, 2012)

O principal objetivo de um Sistema Cooperativo é permitir a comunicação de idéias, compartilhamento de recurso e coordenação de esforços de trabalho. Sua principal meta é permitir o trabalho em conjunto de maneira mais fácil e eficaz, ajudando a colaboração entre os indivíduos envolvidos em um processo, possibilitando que as pessoas envolvidas no projeto tenham uma visão geral do trabalho, permitindo um entendimento compartilhado sobre o andamento das tarefas ou de todo o trabalho. (MOTA; CAVALCANTE, 2009)

Vivemos hoje uma era colaborativa. Podemos constatar isso tudo no que chamamos de Web 2.0. Deixamos de ser agentes passivos para nos tornamos ativos na construção da Web. O conhecimento passa a ser difundido através de colaboração de autores em diversos *wikis*, *softwares* para criação e edição compartilhada de documentos com uso de hipertexto. O debate e a exposição de idéias sobre fatos torna os fóruns espaços ricos em informações. Sistemas de reputação e recomendação em *sites* de comércio eletrônico colaboram para que possamos conhecer hábitos e interesses de diversas regiões ao redor do mundo.

#### 2.1.2 Groupware

A definição atual para *groupware* foi fruto da colaboração de diversas pessoas, dentre elas, Ellis, Gibbs e Rein (1991), que escrevem em seu artigo que a palavra é utilizada para designar um sistema baseado em computador para auxiliar grupos de pessoas a realizarem uma tarefa ou objetivo em comum, provendo uma interface para um ambiente compartilhado.

O termo *groupware* é mais um dos muitos termos utilizados na temática de sistemas colaborativos. Outro termo utilizado é CSCW, um acrônimo para o inglês *Computer*

*Supported Cooperative Work*, que pode ser traduzido como trabalho cooperativo apoiado por computadores. Alguns autores defendem que *groupware* e CSCW possuem o mesmo significado, outros acreditam que *groupware* são sistemas que apoiam o trabalho em grupo proporcionado pelo modelo que engloba o CSCW, ou seja, é um contexto bem mais amplo, onde *groupware* encontra-se incluso (PIMENTEL; FUKS, 2012).

De fato, a forma que abordamos a construção de *softwares* hoje é diferente. Ferramentas que apoiem a colaboração vem sendo amplamente utilizadas, como fóruns e *wikis*. Cada vez mais, gestores utilizam ferramentas computacionais para comunicação e coordenação de suas equipes. Os mensageiros instantâneos e o correio eletrônico são ferramentas indispensáveis em diversas empresas, pois facilitam a comunicação entre os funcionários. Diversos *softwares* de gerência de projetos existem para apoiar equipes de trabalho em seus objetivos. Todas essas tecnologias que conhecemos e utilizamos hoje amplificam a capacidade de seus usuários em concluir tarefas.

### 2.1.3 Modelo 3C

Para um sistema ser considerado colaborativo, ele precisa satisfazer algumas características. Ellis, Gibbs e Rein (1991) enumeram três características que *softwares* voltados ao trabalho em grupo devem possuir: comunicação, cooperação e colaboração. Esses conceitos deram origem ao que conhecemos hoje como Modelo 3C.

De forma geral, trabalhar em grupo é mais vantajoso que trabalhar sozinho. Isso se dá pois ao agregarmos conhecimentos e força de trabalho para terminar uma tarefa nos custa menos tempo e uma parcela menor de esforço. A sensação de se fazer necessário em uma equipe e a avaliação obtida ao se trabalhar em grupo também geram satisfação.

Porém, podemos perceber algumas desvantagens no trabalho coletivo. Trabalhar em grupo sem qualquer orientação ou controle sobre os demais integrantes pode gerar resultados ruins. Sem alguém para coordenar o trabalho, o não cumprimento em tempo hábil do objetivo em comum ou mesmo a falta de soluções para pequenas tarefas necessárias para o objetivo são erros comuns que se pode atingir. A comunicação é essencial, pois a interação e o diálogo entre o grupo é que possibilita que todos estejam cientes do que se tem de fazer. A comunicação permite também a difusão do conhecimento e a busca por soluções para problemas que possam existir durante as diversas etapas para o cumprimento do objetivo (FUKS; RAPOSO; GEROSA, 2002).

Identificou-se certas características comuns ao se trabalhar com sistemas colaborativos de tal forma que se pôde definir um modelo. Se observadas e trabalhadas de forma positiva, o sucesso do trabalho colaborativo é garantido. O Modelo 3C tem por base as características propostas por Ellis, Gibbs e Rein (1991) e descrito em Borghoff e Schlichter (2000), com algumas terminologias ligeiramente utilizadas para significados diferentes.

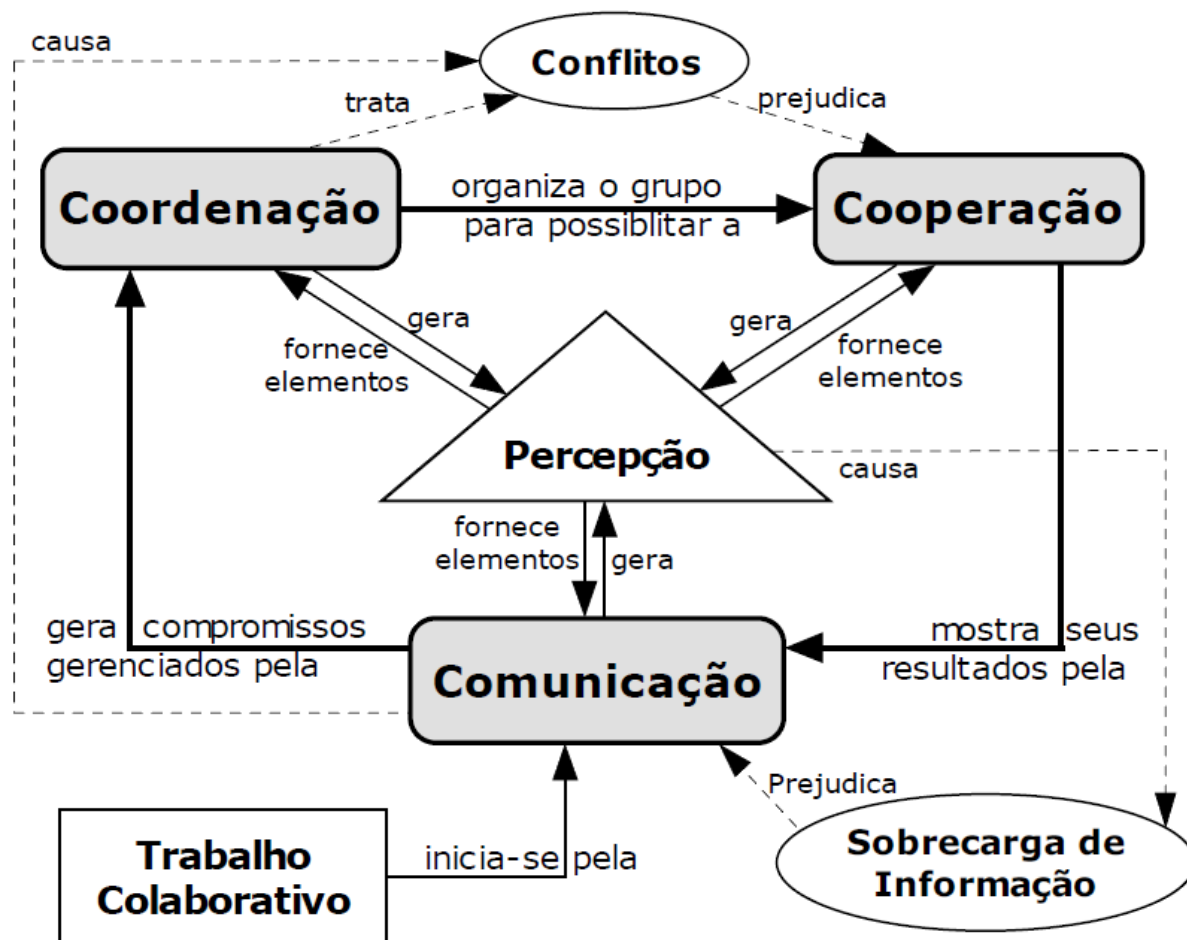


Figura 1 – Representação do Modelo 3C.

Fonte: Fuks, Raposo e Gerosa (2002).

Para ambos, comunicação é a troca de informações entre pessoas e cooperação é a atuação em conjunto de pessoas por um objetivo comum. O termo colaboração foi substituído por coordenação para significar o gerenciamento das pessoas e das etapas necessárias para a realização do objetivo. A Figura 1 ilustra o Modelo 3C, explicitando essas características e os relacionamentos que torna cada uma delas dependentes da outras.

### Comunicação

A comunicação permite o entendimento das pessoas, o compartilhamento de idéias e a difusão do conhecimento entre os membros do grupo. É instrumento necessário pois permite auxiliar na realização de tarefas interdependentes ou mesmo na negociação entre elas.

São inúmeras as ferramentas existentes para facilitar a comunicação. Elas podem estar inseridas em um *software* ou mesmo alheia ao computador. Podem utilizar desde linguagem natural até símbolos e diagramas, desde que todos possam se entender. As ferramentas se distinguem no tempo de reação entre os participante e por isso são classificadas

em síncronas ou assíncronas.

A comunicação com ferramentas síncronas requerem um tempo de resposta menor. São utilizadas quando a comunicação necessita da reação imediata dos participantes, quando se precisa de soluções rápidas para tarefas ou mesmo quando os participantes estão realizando um *brainstorming* de idéias. São exemplos deste tipo de ferramenta os mensageiros instantâneos. Sem a interação do computador, podemos citar o bate-papo em reuniões presenciais ou uso do telefone.

Já as ferramentas assíncronas valorizam a reflexão entre os participantes, de forma a permitir a análise e o pensamento antes de responder às perguntas dos outros participantes. São utilizadas quando o debate permite um tempo maior ou quando se deseja planejar etapas futuras. Exemplos de ferramentas assíncronas são os fóruns e os correios eletrônicos (MOTA; CAVALCANTE, 2009).

### Coordenação

O papel da coordenação é de atuar como mediador entre os integrantes, coordenando atividades e evitando que integrantes possam realizar tarefas exaustivas ou conflitantes. Desta forma, se garante que as tarefas, frutos da comunicação entre os integrantes de um grupo, sejam cumpridas (FUKS; RAPOSO; GEROSA, 2002).

A coordenação pode se valer de diversos métodos para o sucesso da equipe. Organizar etapas, estabelecer funções dos participantes, rever prazos e gerenciar o tempo são exemplos desses métodos. Para isso, se pode utilizar modelos para facilitar o trabalho em grupo, como gráficos, fluxogramas e documentos de padronização.

Cabe ainda ao coordenador resolver conflitos de comunicação ou interesses. Outros tipos comuns de conflitos que podem ocorrer durante um projeto são competição e difusão de responsabilidades (FUKS; RAPOSO; GEROSA, 2002 apud SALOMON; GLOBERSON, 1989).

### Cooperação

A cooperação é a forma de contribuir com o grupo ao compartilhar e registrar informações em um espaço onde todos tenham acesso. O registro de informações tem por fim garantir o entendimento dos integrantes, permitindo a solução de desentendimentos e evitando equívocos.

A cooperação produz conhecimento. Este conhecimento pode ser formal, como, por exemplo, a documentação produzida e preservada ao longo de todo o processo pelos integrantes. Essa documentação de forma catalogada e estruturada gera conforto a toda equipe ao auxiliar na tomada de decisões e conhecimento.

Porém, durante o processo de construção dessa documentação, conhecimentos informais como idéias e pontos de vistas permitem auxiliar no históricos de discussões e tomadas de decisões (FUKS; RAPOSO; GEROSA, 2002).

Diversos instrumentos pode ser utilizados para auxiliar a cooperação entre os integrantes. São exemplos de instrumentos as bibliografias e documentos. De forma eletrônica, temos também os *wikis*.

## 2.2 Sistemas de Cotação de Preços

cotação co.ta.ção sf (cotar+ção) 1 Ação ou efeito de cotar. 2 Com Preço corrente das mercadorias, dos papéis de crédito, títulos da dívida pública etc. 3 Indicação desses preços. 4 Apreço, conceito, conta. (MICHAELIS, 2014)

Cotação de preços em economia é o estabelecimento de valores para produtos, moedas, títulos, ações, etc. Esse valor é geralmente um preço agregado estabelecido em uma unidade de moeda. Esse preço geralmente é estipulado por fatores como porcentagem de lucro sobre o gasto na produção ou a quantidade de procura e oferta pelo produto.

A cotação de preços leva em conta a variação dos valores encontrados para um mesmo produto no mercado. Essa variação de preços pode ocorrer devido a diversos fatores, como mão de obra e a matéria prima utilizadas. Essa variação também gera concorrência, pois no capitalismo, o objetivo ao vender produtos é gerar lucros.

Para quem compra, porém, o objetivo é economizar. Por isso, os compradores tendem a pesquisar preços antes de adquirir seus produtos. Ao pesquisar por produtos ou serviços, gasta-se tempo e esforço para se localizar vendedores apropriados. Isso faz com que métodos como recomendações e anúncios em classificados de jornais e revistas ainda sejam bastante utilizados.

Um sistema de cotação de preços é um sistema computacional capaz de catalogar os mais diversos produtos e serviços e permitir que informações de fornecedores, seus itens e preços possam ser registrados para que potenciais compradores possam decidir de quem comprar, de forma a minimizar seus gastos e garantir a satisfação de adquirir uma boa negociação.

O uso deste tipo de *software* tem por finalidade auxiliar na apuração do menor preço de um determinado produto ou serviço. Pode também funcionar como consulta para relação de produtos e empresas e mostrar análises de mercado seguindo a tendência de preços ao longo de um determinado tempo.

É utilizado principalmente por empresas e pessoas. No caso de empresas, se divulga previamente uma lista de produtos a serem cotados, geralmente essa lista possui um grande



volume de produtos. Assim, pode acontecer que a empresa cote os preços por meio físico ou eletrônico ou os próprios fornecedores já repassem essas informações com base na lista divulgada. No caso de pessoas, elas utilizam as informações já previamente inseridas pelos fornecedores dos produtos e montam sua pesquisa de forma a apurar o menor preço para os produtos escolhidos.

Sistemas de cotação de preços são utilizados devido a suas inúmeras vantagens. Por ser um sistema computacional, o armazenamento de dados é feito de forma digital. Não é preciso uso de papel e outros recursos, somente do dispositivo onde o *software* está instalado. A visualização dos dados tende a ser bem mais agradável. Tarefas como pesquisa se tornam bem mais acessíveis e simples de serem executadas. O esforço também é menor, uma vez que tarefas cansativas como catalogação de produtos e fornecedores são reutilizáveis e de fácil inserção no sistema. Assim, a experiência final se torna mais atrativa e menos cansativa.

Hoje em dia, há diversas soluções no mercado. As mais utilizadas são as soluções *Web*, pois permitem a visualização dos dados em qualquer local, bastando apenas ter um navegador de *internet*. São exemplos de soluções deste tipo Custo Certo (2013) e Dicto Sistemas (2013). Algumas soluções *desktop* ainda são utilizadas hoje, porém com menos frequência. São exemplos o sistema Acca Software (2014) e Cota Fácil<sup>1</sup>.

Todos os *softwares* existentes no mercado atendem os requisitos necessários para um sistema de cotação de preços. O que determina o fator de sucesso destas ferramentas é a quantidade de usuários utilizando-os. Em geral, os sistemas *Web* tem tido maior receptividade pelos usuários, pelas vantagens que eles oferecem. A possibilidade de ter acesso aos dados atualizados em qualquer lugar, bem como poder obter dados atualizados de fornecedores e novos produtos são exemplos de recursos que atraem usuários.

Algumas soluções exploram dispositivos móveis. É o caso dos aplicativos Boa Lista<sup>2</sup>, Meu Carrinho<sup>3</sup> e QQFalta<sup>4</sup>. Por utilizar dispositivos móveis, essas aplicações se tornam mais atraentes e populares. No caso do Meu Carrinho, os dados são armazenados em servidores na *internet* e o usuário pode ver listas de compras de outros usuários, funcionando como uma rede social.

Dentre todas as soluções existentes no mercado brasileiro, até onde sabemos, nenhuma das soluções utilizam dispositivos móveis como *smartphones* com uso de *Near Field Communication* para auxiliar na cotação de preços. Soluções que utilizem esse tipo de recurso no mercado atual destacam-se dentre os demais, uma vez que não necessitariam estar conectados na *internet*, facilitando o compartilhamento das informações utilizando

<sup>1</sup> Disponível em <<http://www.antaresassessoria.com.br/produtost.php?id=7>>. Acessado em 21 de Março de 2014.

<sup>2</sup> Disponível em <<https://www.boalista.com.br/>>. Acessado em 21 de Julho de 2014.

<sup>3</sup> Disponível em <<http://www.meucarrinho.com.br/>>. Acessado em 21 de Julho de 2014.

<sup>4</sup> Disponível em <<http://www.qqfalta.com.br/>>. Acessado em 21 de Julho de 2014.

redes locais.

Assim, percebe-se que sistemas de cotação de preços tendem a participar mais ativamente na vida dos usuários, uma vez que este tipo de *software* alcança também as novas tecnologias que surgem a cada dia no mercado. É importante frisar que, além da proposta desses sistemas de fazer seus usuários economizarem capital e tempo, abordar diversas outras características de forma a dar mais comodidade e facilidade de acesso a seus usuários torna-se um diferencial importante para a popularidade de um sistema de cotação de preços.

## 3 Near Field Communication

Neste capítulo, abordaremos a tecnologia *Near Field Communication*, uma tecnologia de rede sem fio que facilita a comunicação e troca de dados entre dispositivos utilizando rádio-frequência. Trataremos de informações sobre sua criação, grupos e pessoas envolvidos na padronização desta tecnologia, aspectos técnicos, tipos de *tag* e modos de operação.

Veremos como o NFC está mudando a forma de interagir com o ambiente em países que já adotam a tecnologia no cotidiano de seus cidadãos. Abordaremos ainda casos de uso desta tecnologia.

### 3.1 Contextualização

Tecnologias de radiofrequência tem suas origens na época da II Guerra Mundial. Eram utilizadas na identificação de aeronaves, num sistema chamado IFF (*Identify Friend or Foe*). Esse sistema consistia em detectar se uma aeronave era amiga ou inimiga, enviando um sinal para a outra aeronave e recebendo uma resposta. Caso não houvesse resposta, a nave era identificada como inimiga (WIKIPEDIA, 2014; BILGINER; LJUNGGREN, 2011).

Um sistema RFID (do acrônimo em inglês *Radio-frequency identification*) funciona de forma parecida, onde temos dois *transponders* (abreviação do inglês *transmitter-responder*). Um *transponder* é um dispositivo eletrônico com função de receber e enviar sinais em uma determinada radiofrequência. Há dois tipos de *transponders* no sistema RFID. O *transponder* ativo, também chamado de transmissor ou emissor, se encarrega de enviar o sinal e receber o sinal de resposta. O *transponder* passivo, também chamado de receptor ou *tag*, só é capaz de responder a um sinal do emissor após ser estimulado pela radiofrequência do mesmo. Por vezes, um agente intermediário, denominado *Middleware*, é utilizado. O *Middleware* é geralmente um *software* encarregado de filtrar informações, realizar o compartilhamento de dados com outros *softwares* e gerenciar a infraestrutura do sistema. A Figura 2 ilustra os principais componentes de um sistema RFID.

Sistemas RFID são muito utilizados em nosso dia a dia. Muitas lojas utilizam sistemas de segurança baseados em RFID. Os produtos possuem um dispositivo (chamado *tag*) onde se armazena um *bit*, que pode ter valor 0 para quando a mercadoria não foi paga e 1 para quando a mercadoria foi paga. Ao passar em um dispositivo geralmente colocado nas portas das lojas, um sinal é enviado e esse *bit* é lido, verificando se a mercadoria está paga. Há também sistemas de estoques todos controlados por RFID, onde os produtos

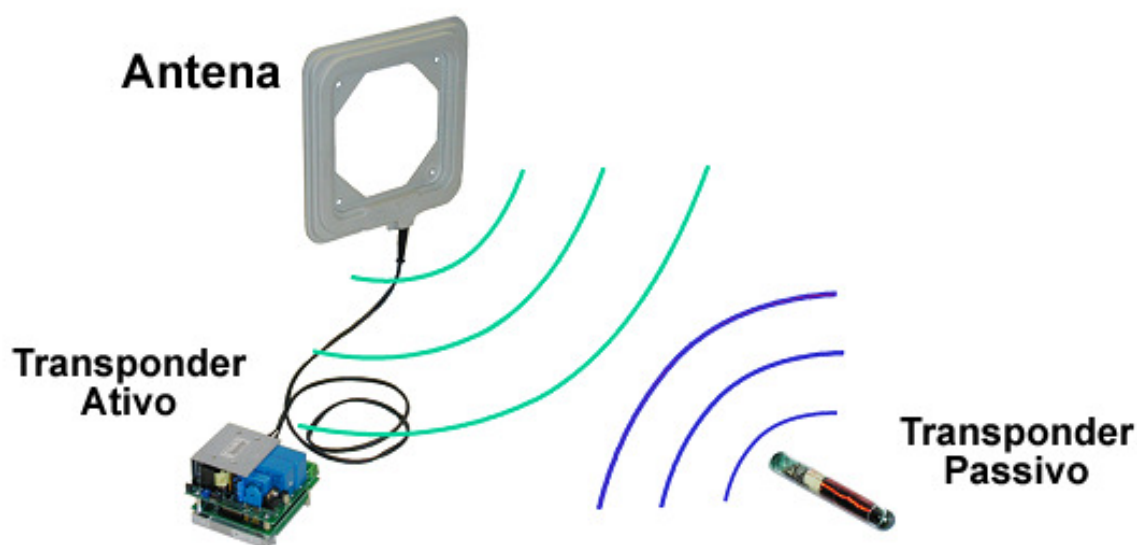


Figura 2 – *Transponders* Ativos e Passivos - Sistema RFID

Fonte: EBV Elektronik (2010). Traduzida pelo autor.

são catalogados e etiquetados, e um dispositivo leitor é utilizado para fazer a leitura e contabilização dos produtos em estoque de uma loja. Estruturas de sistemas RFID para estoque podem ser adaptadas para sistemas de organização de documentos ou livros, como sistemas de recursos humanos e bibliotecas.

NFC é uma tecnologia baseada em RFID, onde seu diferencial está na distância de comunicação entre os dispositivos. Os sistemas atuais baseados em RFID pretendem abranger áreas para varredura das informações em *tags* de forma rápida. NFC foi concebido para que a leitura de informações seja uma ação voluntária entre os dispositivos envolvidos, por isso a distância para comunicação não deve passar de 10 centímetros.

NFC está padronizada na ISO/IEC 18092, com última revisão em 2013. Este padrão foi desenvolvido prevendo suporte a sistemas baseados em RFID, como cartões inteligentes. Por ser um padrão na indústria, muitas aplicações e dispositivos já foram desenvolvidos com suporte à tecnologia NFC. NFC vem sendo amplamente utilizada em sistemas de pagamento, saúde e transporte.

## 3.2 NFC Forum

Em 2002, Sony e Phillips iniciaram os trabalhos com NFC. Em 2004, com a entrada da Nokia aos trabalhos, fundaram o NFC Forum com o objetivo de avançar no uso do NFC, desenvolvendo especificações e educando o mercado sobre o uso do NFC (NFC Forum, 2014). Atualmente com 190 membros, o NFC Forum organiza eventos, reúne diversas informações e tem papel crucial na divulgação da tecnologia NFC.

Após quase 10 anos de existência, membros como Nokia, Samsung, LG e Sony tem apoiado o uso do NFC ao lançar produtos de sua linha de dispositivos móveis (*smartphones* e *tablets*) com *hardware* NFC. Com isso, diversas aplicações foram desenvolvidas para poder usufruir dessa funcionalidade nesses dispositivos. Visa e MasterCard tem apoiado a iniciativa também ao utilizar os dispositivos NFC como cartões, criando assim novos mecanismos para pagamentos e transações monetárias.

Todas essas empresas cada vez mais inserem na vida das pessoas a tecnologia NFC, e acredita-se que esse é o ponto-chave do sucesso do NFC (WANT, 2011). De fato, as pessoas começam então a descobrir como funciona o NFC e sentem-se tentadas a usufruir dessa tecnologia.

### 3.3 Aspectos Técnicos

NFC é uma extensão do padrão de cartões inteligentes de identificação (ISO/IEC 14443), sendo assim, compatível com os cartões inteligentes existentes no mercado. Como esses cartões inteligentes são baseados em tecnologia de radiofrequência, a tecnologia NFC opera na faixa de rádio HF (do inglês *High Frequency*, traduzido por alta frequência) de 13.56 MHz. NFC é uma tecnologia sem fio de curto alcance, geralmente havendo comunicação em uma distância de 10 centímetros ou menos. Sua taxa de velocidade varia entre 106 Kbit/s a 424 Kbit/s, dependendo do modo utilizado. Essa velocidade é menor que em outras tecnologias existentes já consolidadas, como Bluetooth<sup>®</sup> e Wi-Fi<sup>™</sup>.

NFC só é possível por causa de um princípio físico chamado acoplamento indutivo. Quando uma corrente elétrica flui através de um fio, ela gera um campo magnético. Este campo magnético pode ser amplificado a medida que aproximamos os fios. Por exemplo, em uma bobina quanto mais enrolamos o fio, maior é o campo magnético gerado. Ao aproximarmos dois campos magnéticos, eles se tornam um só campo, fluindo de um pólo a outro, como se cada pólo fosse uma antena que capta o campo magnético do outro. Para que este fenômeno ocorra, as duas bobinas devem estar bem próximas (BILGINER; LJUNGGREN, 2011; YAQUB; SHAIKH, 2012). O acoplamento indutivo está ilustrado na Figura 3, onde podemos perceber como se comporta o campo magnético quando um cartão inteligente, um dispositivo receptor, se aproxima de um leitor de cartões inteligente, um dispositivo transmissor.

No sistema RFID, denomina-se antena a bobina de fios usada para captar o campo magnético de uma outra bobina de fios, quando em estado de acoplamento indutivo. Como NFC funciona na faixa HF do espectro de rádio do RFID, todo seu *hardware* e dispositivos funcionam de forma análoga. Em um dispositivo NFC, encontraremos uma antena que fica emitindo seu campo magnético e é capaz de detectar um estado de *loop* (em física, significa quando temos um comportamento do campo semelhante a um aro ou anel) com outra

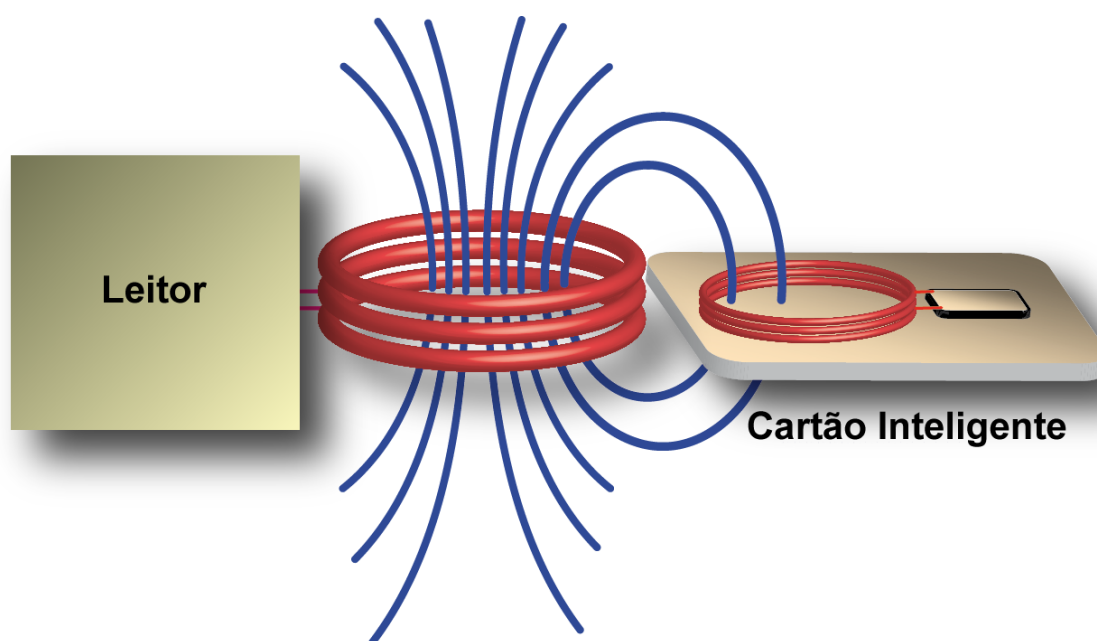


Figura 3 – Princípio de Funcionamento do NFC: Acoplamento Indutivo.

Fonte: Disponível em <<http://rfid-handbook.de>>. Traduzida pelo autor.

antena de um segundo dispositivo NFC. Assim, a comunicação entre eles é estabelecida e os dados são enviados de um para o outro.

Semelhante ao RFID, um dispositivo NFC pode ter forma de interação ativa ou passiva. Um dispositivo ativo possui uma fonte de energia. Sua antena é maior para poder produzir energia suficiente para ativar um dispositivo passivo. São exemplos de dispositivos ativos *smartphones* e leitores de NFC. Um dispositivo passivo não possui fonte de energia, por isso, quando um dispositivo ativo se aproxima, ele utiliza o campo magnético emitido do dispositivo ativo para funcionar. Assim, os dados são recuperados do dispositivo passivo para o dispositivo ativo. Um dispositivo passivo é sempre menor e mais barato que o dispositivo ativo. São exemplos de dispositivos passivos as *tags* RFID/NFC e os cartões inteligentes (EBV Elektronik, 2010).

NFC opera em três modos básicos de operação. Esses modos se diferem quanto a finalidade pretendida e quanto a forma de interação.

- a) Emulação de cartão: Este modo de operação serve para garantir compatibilidade com as tecnologias de cartões RFID que já existem no mercado. Ao aproximar um dispositivo NFC, geralmente *smartphone*, a um leitor de cartão inteligente, o dispositivo funciona de forma análoga a um cartão inteligente.
- b) Leitor/Escritor: Este modo de operação é utilizado principalmente para a leitura e escrita de dados em *tags* RFID. No modo leitor, é possível recuperar dados de *tags*, geralmente associadas a uma aplicação. O modo escritor serve para escrever dados



Figura 4 – Representação do suporte do NFC à quatro tipos de *tags*.

Fonte: Disponível em <<http://www.sony.net/Products/felica/NFC/forum.html>>. Traduzida pelo autor.

em *tags* para que se possa ler essas informações posteriormente. Dentre as aplicações que usam este modo estão os pôsteres inteligentes e compras remotas.

- c) *Peer-to-peer*: Este modo de operação habilita a comunicação entre dois dispositivos ativos. Tem por objetivo compartilhar dados entre esses dispositivos. Assim, é possível que ambos os dispositivos possam tomar decisões para a troca de dados. Quando um dispositivo ativo entra no modo de operação *peer-to-peer*, os dados são enviados sobre um canal bidirecional (*full-duplex*). Isso significa que enquanto um está transmitindo os dados, o outro está escutando e iniciará sua transmissão de dados assim que o primeiro terminar. Possíveis aplicações são transferências monetárias e redes sociais (DESAI; SHAJAN, 2012).

Há quatro tipos de *tags* documentadas na especificação do NFC, que são especificações de *tags* já usadas em sistemas de RFID (ISO/IEC 14443 tipos A e B e Sony<sup>©</sup> FeliCa<sup>™</sup>). A Figura 4 demonstra o suporte a esses quatro tipos. As *tags* se diferenciam quanto a finalidade de uso, capacidade e taxa de transferência. A mais lenta, o Tipo 1, transfere dados na velocidade de 106 Kbit/s e armazena apenas 96 *bytes*. A mais rápida, o Tipo 4, transfere dados a uma taxa de velocidade de até 424 Kbit/s, e sua capacidade de armazenamento pode variar até 32 *Kbytes*. As *tags* do Tipo 1 e 2 podem ser reescritas diversas vezes, mas há uma forma de mudá-las para somente leitura, caso seja necessário. Já as *tags* do tipo 3 e 4 são do tipo somente-leitura, logo são escritas somente uma vez (CHANDLER, 2014).

Segue abaixo uma descrição mais detalhada com os quatros tipos de *tags*:

- **Tipo 1:** Baseada no padrão ISO/IEC 14443 tipo A, esta *tag* tem capacidade de reescrita, porém o usuário tem a capacidade de configurá-la em modo somente-leitura. Tem capacidade de armazenamento de 96 *bytes* para dados de usuário e 120 *bytes* de capacidade total. A velocidade de transferência é de 106 Kbit/s, suficiente para enviar os dados de forma que apenas uma rápida aproximação entre os dispositivos transfira os dados. Este tipo de *tag* é ideal para pequenas quantidade de informação, como URLs ou dados de configurações para tomada de decisão de algum *software*. É uma *tag* de fácil acesso e baixo custo. São exemplos de soluções existentes no mercado compatíveis com este tipo de *tag*: Innovision<sup>©</sup> Topaz, Broadcom<sup>©</sup> BCM20203.
- **Tipo 2:** Baseada no padrão ISO/IEC 14443 tipo A, esta *tag* é semelhante à do tipo 1 no que diz respeito a capacidade de reescrita e modo somente-leitura. Porém, sua capacidade de armazenamento é de apenas 48 *bytes* para dados de usuário e 64 *bytes* de capacidade total. A velocidade de transferência é de 106 Kbit/s. Sua principal vantagem em relação ao Tipo 1 está no suporte a tratamento de colisão de dados. São exemplos de soluções existentes no mercado compatíveis com este tipo de *tag*: NXP<sup>©</sup> MIFARE<sup>™</sup> Ultralight.
- **Tipo 3:** Esta *tag* é baseada na especificação do sistema Sony<sup>©</sup> FeliCa<sup>™</sup>. Parte dessa especificação foi incorporada na especificação do NFC (ISO/IEC 18092), uma vez que o padrão FeliCa<sup>™</sup> foi proposto como ISO/IEC 14443 Tipo C mas foi rejeitado. Diferentemente dos tipos anteriores, esta *tag* pode vir pré-configurada de fábrica para funcionar em modo escrita ou somente-leitura. O Tipo 3 não especifica uma capacidade padrão, mas sim uma arquitetura de gerenciamento de memória. Cada bloco da memória deve ter um valor fixo de 16 *bytes*. A quantidade de blocos deve variar dependendo do *hardware*. Serviços são encarregados de referenciar os blocos de memória. Em teoria, cada Serviço pode referenciar até 1 Mb de memória. Pode haver diversos Serviços em uma mesma *tag*. A velocidade de transferência pode ser 212 Kbit/s ou 424 Kbit/s. Como recomendado na especificação do NFC, este tipo de *tag* é em geral utilizado em aplicações complexas. Também possuem um alto custo. São exemplos de soluções existentes no mercado compatíveis com este tipo de *tag*: Sony<sup>©</sup> FeliCa<sup>™</sup> (NOKIA, 2014a).
- **Tipo 4:** Esta *tag* é compatível com os padrões ISO/IEC 14443 tipo A e tipo B. Pode vir pré-configurada de fábrica para funcionar em modo escrita ou somente-leitura. A capacidade total de armazenamento pode variar. Sua arquitetura de gerenciamento de memória é parecida com *tags* do Tipo 3, porém os Serviços podem ter espaços alocados de até 32 *Kbytes*. Suporta velocidade de transferência de 106, 212 ou 424



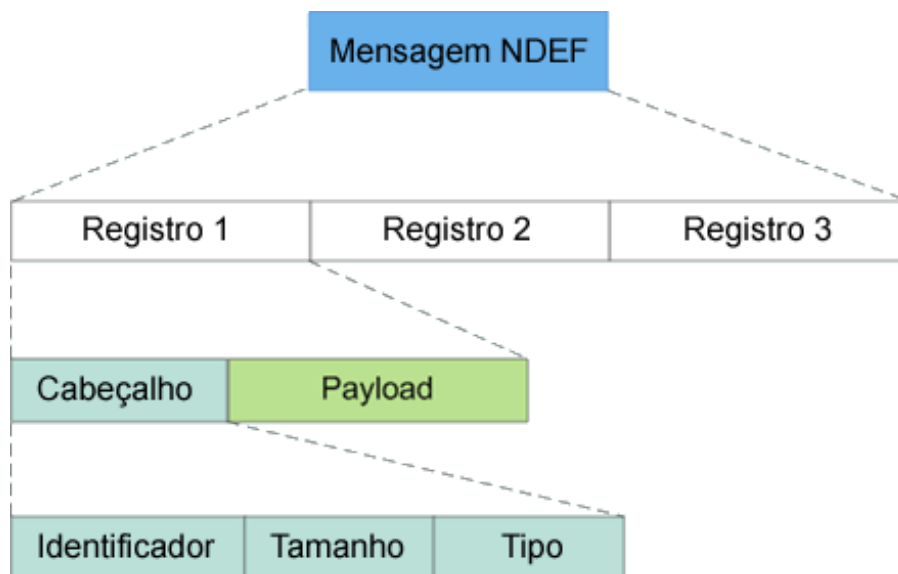


Figura 5 – Representação simplificada do formato NDEF.

Fonte: Nokia (2014b). Traduzida pelo autor.

Kbit/s. São exemplos de soluções existentes no mercado compatíveis com este tipo de *tag*: NXP<sup>©</sup> DESFire, SmartMX-JCOP (NOKIA, 2011; NFC Forum, 2014).

Quando as aplicações NFC leem dados armazenados em *tags* ou de outros dispositivos, estão na verdade lendo dados no formato *NFC Data Exchange Format* (NDEF, traduzido livremente por Formato de Troca de Dados NFC). Descrito na especificação do NFC, o NDEF define um formato de encapsulamento de mensagens para troca de informações (NOKIA, 2014b).

Uma mensagem NDEF é composta por um ou mais registros (*records*) NDEF. A quantidade de registros em uma mensagem NDEF dependem da aplicação e do tipo de *tag*. Cada registro contém um *payload* (conteúdo de dados). Vários registros podem ser encadeados para dar suporte a *payloads* com maiores volumes de dados. Além disso, um registro NDEF contém três parâmetros sobre o *payload*: o tamanho, o tipo e um identificador. Este formato pode ser visualizado de forma simplificada na Figura 5.

O campo Identificador, um campo opcional, serve para que um *payload* possa referenciar outros *payloads*, em uma estrutura baseada em URI (do acrônimo em inglês *Uniform Resource Identifier*). O campo Tamanho indica o tamanho do *payload* no registro NDEF. O conteúdo do campo Tipo é especificado pelo campo TNF (*Type Name Format*, traduzido livremente por Formato do Nome do Tipo). Este campo é um dado informado no primeiro octeto do registro NDEF, representado pela Figura 6. Os possíveis valores que podem ser utilizados no campo TNF estão descritos na Tabela 1 (NFC Forum, 2006).

Cada *bit* do octeto tem seu significado descrito na lista abaixo:

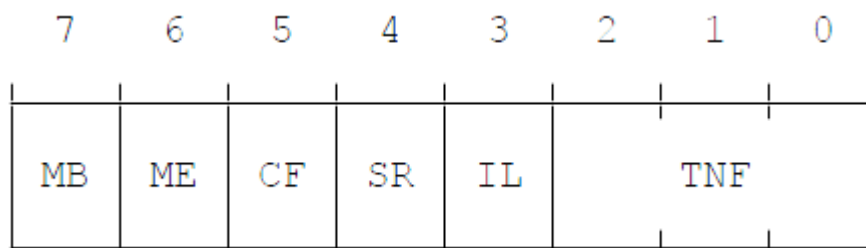


Figura 6 – Estrutura do primeiro octeto de um registro NDEF.

Fonte: (NFC Forum, 2006)

- **Message Begin (MB)**: Contém o *bit* 1 se este for o primeiro registro da mensagem NDEF.
- **Message End (ME)**: Contém o *bit* 1 se este for o último registro da mensagem NDEF.
- **Chunk Flag (CF)**: Contém o *bit* 1 se este for um pedaço de um *payload* que será transmitido em vários registros.
- **Short Record (SR)**: Se for configurado com o *bit* 1, isso significa que o campo Tamanho do *payload* contém apenas um octeto. Caso configurado com o *bit* 0, o campo Tamanho do *payload* conterá quatro octetos (registro normal). Esta *flag* é destinada ao encapsulamento de *payloads* que se encaixam dentro do campo Payload com tamanho entre 0 e 2 *Kilobits*.
- **Identification Length (IL)**: Se for configurado com o *bit* 1, indica que os campos Tamanho do ID e ID estão preenchidos com dados.
- **Type Name Format (TNF)**: Os últimos três *bits* são reservados ao campo TNF para especificar o valor do campo Tipo.

### 3.4 Comparação com outras Tecnologias

NFC é uma tecnologia nova no cotidiano das pessoas. Algumas pessoas são aversas à novas tecnologias, e por isso deixam de experimentá-la por saber que no mercado existem outras soluções para a comunicação sem fio. A proposta do NFC não é concorrer com essas tecnologias existentes, o que o NFC realmente faz é melhorar a forma como essas tecnologias são utilizadas.

Nesta seção, iremos comparar as tecnologias sem fios existentes com o NFC em aspectos como objetivo, tipo de rede, alcance de rede, velocidade de transferência, consumo

Tabela 1 – Valores para o campo TNF.

Formato do Nome do Tipo	Valor
Vazio	0x00
Tipos conhecidos do NFC Forum (NFC RTD)	0x01
Tipos de Mídia como definidos no RFC 2046	0x02
URIs como definidas no RFC 3986	0x03
Tipos Externos do NFC Forum	0x04
Desconhecido	0x05
Sem Mudanças	0x06
Reservado	0x07

Fonte: (NFC Forum, 2006)

Tabela 2 – Tabela de Comparação entre Tecnologias de Rede.

Tecnologia	Tipo de Rede	Alcance	Velocidade de Transferência	Consumo de Energia	Nível de Segurança
NFC	PAN	> 10 cm	Até 424 Kbit/s	Baixo	Alto
Bluetooth®	PAN	> 3 m	A partir de 2.1 Mb/s	Baixo	Médio
Wi-fi™	WLAN	> 100 m	A partir de 54 Mb/s	Alto	Médio
ZigBee®	PAN	> 100 m	250 Kbit/s	Baixo	Alto
iRDA	PAN	> 2 m	Até 115.2 Kbit/s	Baixo	Alto

Fonte: Bluetooth SIG (2014), Wi-Fi Alliance (2014), ZigBee Alliance (2014), Infrared Data Association (2014), NFC Forum (2014).

de energia e nível de segurança. Podemos ainda observar a Tabela 2, que resume as principais características de cada uma das tecnologias.

### 3.4.1 Bluetooth

O Bluetooth® é uma tecnologia presente na maioria dos *smartphones* e *tablets*. Tem como objetivo prover uma forma de conectar e trocar informações entre diversos tipos de dispositivos em curta distância. Assim como o NFC, foi construída para ser uma rede PAN (*Personal Area Network*). Rede PAN é um tipo de rede de computadores que conecta dispositivos em uma pequena área, com base na localização pessoal do indivíduo.

O alcance do Bluetooth® é de até 3 metros, enquanto NFC possui alcance de até 10 centímetros. Utilizando o NFC, um usuário precisa aproximar os dispositivos para que a conexão seja estabelecida. Isso garante que os usuários tenham realmente a intenção de estabelecer a comunicação. Além disso, NFC é uma rede de comunicação ponto-a-ponto, o que significa que temos apenas dois dispositivos se comunicando por vez. O Bluetooth® foi projetado também para trabalhar como uma rede multi-pontos, onde existe a comunicação

entre vários dispositivos simultaneamente.

O Bluetooth<sup>®</sup> em sua versão 2.0 + EDR (do acrônimo em inglês *Enhanced Data Rate*) possui taxa de transferência de até 2.1 Mb/s, enquanto NFC pode atingir até 424 Kbit/s. A velocidade de transferência do Bluetooth<sup>®</sup> é maior uma vez que o objetivo é a troca de arquivos entre os dispositivos. A proposta do NFC é o compartilhamento rápido de dados, por isso eles, em geral, são menores. NFC foi concebida para facilitar pagamentos, garantir acesso e compartilhar *links* e textos. Assim, a velocidade de transferência do NFC não chega a ser uma desvantagem observando sua proposta (AGRAWAL; BHURARIA, 2012).

O NFC e o Bluetooth<sup>®</sup> podem trabalhar em conjunto para oferecer uma melhor experiência ao usuário. Por exemplo, um dos problemas relatados por algumas pessoas é a quantidade de etapas necessárias para o pareamento Bluetooth<sup>®</sup>. Uma aplicação Bluetooth<sup>®</sup> pode fazer uso do NFC para simplificar este pareamento.

Imaginemos uma aplicação que troca arquivos de vídeo. Para o NFC, com sua baixa taxa de transferência e sua necessidade de manter os dispositivos próximos, é uma tarefa árdua. Porém, se a negociação do pareamento Bluetooth<sup>®</sup> fosse realizada utilizando o NFC, bastaria aproximar os dispositivos para que eles pareassem e uma comunicação Bluetooth<sup>®</sup> fosse estabelecida, iniciando assim a transferência do arquivo de vídeo.

O consumo de energia em ambas as tecnologias é baixo. A versão 4.0 do Bluetooth<sup>®</sup> opera em níveis de energias muito baixos, pois foi projetada para executar em dispositivos bem pequenos, com baterias de poucas células. NFC opera também com baixo consumo de energia. Vale destacar novamente que dispositivos NFC passivos não necessitam de energia, uma vez que eles absorvem a energia do campo de radiofrequência gerado pelos dispositivos ativos.

Bluetooth<sup>®</sup> oferece alto grau de segurança, que a cada versão apresenta novas soluções neste aspecto. Porém, ele já foi alvo de muitas discussões acerca de sua confiabilidade. O que se pode inferir, de acordo com as características do NFC e do Bluetooth<sup>®</sup>, é que a distância menor e a arquitetura de rede ponto-a-ponto promovem ao NFC um grau de segurança maior em áreas com muitas pessoas (KUMAR, 2011).

### 3.4.2 Wi-Fi

Wi-Fi<sup>™</sup>, nome popularizado das redes padrões IEEE 802.11 a/b/g/n, é uma alternativa às redes *Ethernet*, porém sem a utilização de cabos entre os dispositivos. É uma tecnologia já bastante difundida e consolidada, presente em diversos dispositivos eletrônicos como *smartphones*, *tablets*, *consoles*, televisores, *notebooks*, etc. Seu tipo de rede é do tipo WLAN (*Wireless Local Area Network*), que conecta, sem o uso de fios, dispositivos dispostos em uma área limitada, como residências, empresas, escolas, etc.

O alcance das redes Wi-fi<sup>TM</sup> é de até 100 metros, dependendo do padrão utilizado (a, b, g ou n). Esse alcance pode ser aumentado com o uso de dispositivos que repetem o sinal da rede. Wi-Fi<sup>TM</sup> permite a comunicação entre diversos dispositivos simultaneamente (comunicação multi-ponto). Possui velocidades que variam de 54 Mb/s até 300 Mb/s em redes com tecnologias mais recentes.

A interoperabilidade entre o NFC e o Wi-Fi<sup>TM</sup> se dá da mesma forma que NFC interage com o Bluetooth<sup>®</sup>: auxiliando na configuração inicial de comunicação entre os dispositivos. Com NFC, podemos compartilhar facilmente todas as informações de conexão de uma rede Wi-Fi<sup>TM</sup> para um dispositivo (NFC Forum, 2014).

Redes Wi-fi<sup>TM</sup> possuem um alto consumo de energia. Isso acontece por diversos motivos, como a faixa de frequência utilizada, a necessidade de rápida transmissão de dados e a abrangência da área coberta. Geralmente, os dispositivos das redes Wi-fi<sup>TM</sup> estão ligados diretamente à rede de energia elétrica.

Em aspectos de segurança, redes Wi-fi<sup>TM</sup> estão sujeitas aos mesmos tipos de ataques de redes *Ethernet*, como o acesso não autorizado à rede. O acesso à rede Wi-fi<sup>TM</sup> é geralmente feito através do uso de senhas de acesso, que podem ser facilmente quebradas quando não são utilizadas técnicas de criptografia ou quando se utiliza técnicas inadequadas.

### 3.4.3 ZigBee

ZigBee<sup>®</sup> é uma tecnologia de rede sem fio baseada no padrão IEEE 802.15. Foi projetada para uso em redes PAN, com características como baixa taxa de transferência, baixo consumo de energia e segurança. É uma tecnologia mais barata e simples que outras tecnologias como Wi-Fi<sup>TM</sup> e Bluetooth<sup>®</sup> (ZigBee Alliance, 2014).

A principal diferença entre NFC e ZigBee<sup>®</sup> está na proposta de cobertura da área. NFC trabalha com comunicação em curtas distâncias, promovendo o contato intencional entre os dispositivos. Dispositivos ZigBee<sup>®</sup> tem por características transmitir dados sobre longas distâncias, cobrindo até 100 metros. Esse alcance pode ser ainda maior com o uso de malhas de redes, uma vez que para alcançar um determinado ponto, um ou mais dispositivos podem servir como repetidores para que a mensagem chegue a seu destino. Assim, redes ZigBee<sup>®</sup> podem ser caracterizadas como ponto-a-ponto ou ponto-a-multiponto.

A taxa de transferência entre dispositivos ZigBee<sup>®</sup> é definida em 250 Kbit/s. Esta velocidade, considerada baixa, é suficiente para o envio de sinais periódicos ou dados intermitentes. Dessa forma, ZigBee<sup>®</sup> vem sendo amplamente utilizado em aplicações para auxiliar o monitoramento e controle industrial, redes de sensores, automação industrial e de residências (BAKER, 2005).

Redes ZigBee<sup>®</sup> foram projetadas para trabalharem com conexões criptografadas e chaves simétricas. Em geral, é uma tecnologia segura, por ter em sua especificação e

implementação métodos e políticas de segurança.

#### 3.4.4 iRDA

A tecnologia iRDA, conhecida por usar luz infravermelha para comunicação, é bem popular em diversos aparelhos eletrônicos que utilizam controle remoto. Sua idéia é simples: apontar um dispositivo e enviar as informações para outro com um simples comando como um apertar de botão. O usuário sente assim que possui total controle sobre o equipamento (Infrared Data Association, 2014).

É uma tecnologia de rede do tipo PAN. Assim, o alcance do iRDA é de até 2 metros. Esse alcance é influenciado também pelo ângulo entre o transmissor e o receptor. Em geral, para que haja a comunicação, é necessário estar numa posição de até 30 graus com o receptor. Além disso, essa comunicação se dá ponto-a-ponto, uma vez que o feixe de luz será direcionado apenas para um receptor por vez.

A velocidade de transferência depende da intensidade do feixe de luz. Assim, ela varia entre 9.6 Kbit/s e 115.2 Kbit/s. Novas implementações do protocolo iRDA ainda estão sendo desenvolvidas. Essas novas implementações tendem a melhorar a velocidade de transferência, como por exemplo o IrSimple, que consegue velocidades 4 a 10 vezes superiores.

É uma tecnologia de baixo custo e baixo consumo de energia, semelhante ao NFC. Porém, não houve popularização desta tecnologia para o compartilhamento de dados entre *smartphones* e *notebooks*. Acredita-se que a necessidade de posicionar os dispositivos de forma que o transmissor esteja em um determinado ângulo para que o receptor capte a informação seja uma das causas.

iRDA é mais comumente utilizada em locais onde tecnologias baseadas em rádio não podem ser utilizadas, uma vez que existem ambientes, geralmente industriais, que causam interferência e ruídos em redes de comunicação baseadas em rádiofrequência.

É uma tecnologia considerada segura por causa de suas características. Os dados transmitidos por iRDA são difíceis de interceptar devido a direcionalidade do feixe de luz. Além disso, ao utilizar iRDA, o usuário tem total controle sobre para onde vão seus dados e quem vai recebê-los.

### 3.5 Segurança

Conhecer aspectos de segurança é importante ao se trabalhar com uma nova tecnologia. Para os desenvolvedores, criar aplicações robustas e seguras é um diferencial num mundo onde novas aplicações surgem a cada instante. Como a proposta do NFC baseia-se principalmente no pagamento digital e na troca de informações que podem ter

certo grau de privacidade, é importante analisar os diferentes ataques que um dispositivo NFC pode sofrer.

A comunicação entre dispositivos NFC é de curto alcance, o que gera uma segurança em essência. Além disso, o NFC tem funcionalidades nativas para dar suporte a aplicações seguras (NFC Forum, 2014). Desse modo, o NFC promove um notório grau de segurança, uma vez que uma pessoa maliciosa terá certa dificuldade para interceptar a comunicação entre dois dispositivos por causa da proximidade.

De acordo com a ISO/IEC 18092, comunicações NFC não são encriptadas em função da retrocompatibilidade com outras tecnologias RFID. Isso não significa que futuras aplicações não possam encriptar seus dados, fazendo disso uma boa prática (CURRAN; MILLAR; GARVEY, 2012).

## Eavesdropping

*Eavesdropping* é uma técnica de invasão onde um atacante intercepta mensagens, analogamente a um grampo de telefone. Esta técnica é um pouco difícil de ser aplicada em uma comunicação NFC, pois os dispositivos necessitam estar próximos para que ocorra a interceptação.

A principal questão é quão perto um atacante deve estar para que consiga, com sucesso, interceptar as mensagens. Haselsteiner e Breitfuß (2006) enumeram em seu artigo “Security in near field communication” condições para que se consiga mensurar a eficiência do ataque.

- a) Características do campo de radiofrequência do dispositivo emissor, como geometria da antena;
- b) Características da antena do atacante;
- c) Qualidade do receptor e do decodificador de sinal de radiofrequência do atacante;
- d) Ambiente onde o atacante está localizado (exemplo: se está por trás de uma parede ou em um ambiente com ruídos);
- e) Energia utilizada pelo dispositivo NFC.

Como se pode perceber, são inúmeros os fatores para se garantir com sucesso um ataque de *eavesdropping*. Há mais variáveis em questão quando o dispositivo interage de forma ativa ou passiva. O dispositivo ativo gera seu próprio campo de radiofrequência, neste caso sendo possível interceptar mensagens a uma distância de 10 metros. Dispositivos passivos não geram nenhum campo de radiofrequência pois se utilizam do campo de radiofrequência de um dispositivo ativo, sendo assim mais difícil interceptar mensagens,

de modo que a distância para um ataque com sucesso cai para apenas 1 metro (CURRAN; MILLAR; GARVEY, 2012).

## Man-in-the-middle

Neste tipo de ataque, o atacante precisa interceptar e garantir que as mensagens enviadas de um dispositivo NFC a outro não sejam trocadas entre si, mas somente a partir dele. Para ilustrar, imaginemos que Alice e Bob queiram trocar informações via NFC. Ted, um atacante, deseja interceptar as mensagens e se fazer passar por Alice ou Bob. Precisamos analisar como o NFC funciona para verificar a possibilidade de tal fato.

Analisemos primeiro a interação entre dispositivos ativos e passivos. Alice possui um *smartphone*, um dispositivo ativo, e Bob possui uma *tag*, um dispositivo passivo. Ted precisa interceptar as mensagens de Alice e garantir que a comunicação entre Alice e Bob não ocorra. Caso ocorra uma perturbação no campo de radiofrequência do *smartphone* de Alice, facilmente o dispositivo dela detectaria um problema. Além disso, Ted precisa gerar um campo de radiofrequência para que a *tag* de Bob receba a mensagem. Uma vez que já haverá um campo de radiofrequência criado pelo dispositivo de Alice, é tecnicamente impossível que Ted se alinhe com o campo dela. Assim, a *tag* de Bob não entenderia a mensagem enviada por Ted (KUMAR, 2011).

Agora, examinemos a interação entre dispositivos ativos. Alice e Bob possuem ambos um *smartphone*. Neste caso, a detecção de um intruso é ainda mais fácil. Ted, ao perceber que Alice está enviando uma mensagem para Bob, precisa interceptar e perturbar o campo gerado pelo dispositivo dela. Neste momento, o dispositivo de Alice perceberia um problema. Imaginemos que o *smartphone* dela não detectou o problema e agora estaria esperando a mensagem de Bob. Ted ainda precisa enviar a mensagem para Bob. Como tanto o *smartphone* de Alice e Bob estão aguardando uma mensagem, ambos receberiam a mensagem de Ted. Assim, novamente, o dispositivo de Alice perceberia que há um problema (HASELSTEINER; BREITFUSS, 2006).

Desse modo, é fácil concluir que ataques *Man-in-the-middle* são praticamente impossíveis de ocorrer e, caso ocorressem, seriam de fácil detecção.

## Corrupção de Dados

Este tipo de ataque preocupa-se em corromper os dados trocados entre dois dispositivos. Como o NFC utiliza campos de radiofrequência, este tipo de ataque pode ser reproduzido facilmente perturbando os sinais e corrompendo dados. Porém, caso as mensagens não sejam trocadas, os dispositivos NFC podem parar a atual comunicação para estabelecer uma nova quando for propício. Este tipo de ataque é mais parecido com Ataques de Negação de Serviço (DoS, acrônimo do inglês *Denial of Service*) e não gera



riscos de roubo de informação ou manipulação de dados (CURRAN; MILLAR; GARVEY, 2012).

### Inserção de Dados

Ataques de inserção de dados em redes de computadores são realizados quando um atacante insere mensagens durante a comunicação entre dois dispositivos. Para que este tipo de ataque possa ocorrer, é importante que o dispositivo transmissor demore um tempo para emitir a mensagem. Assim, um atacante poderia enviar mensagens antes que o dispositivo transmissor. Porém, este tipo de ataque pode ser facilmente detectado, uma vez que o dispositivo receptor espera uma mensagem do dispositivo transmissor. Ele detectaria que recebeu uma mensagem de um terceiro dispositivo, o atacante. Ainda há a possibilidade de que o atacante emita a mensagem no mesmo momento que o transmissor envia sua mensagem, fazendo com que ocorra uma sobreposição dos campos de radiofrequência e, conseqüentemente, corrupção dos dados (IMHONTU; KUMAH, 2010).

### Modificação de Dados

O objetivo deste tipo de ataque é fazer com que um dispositivo receptor receba uma mensagem modificada. Para isso, é preciso que o atacante, na hora em que os dados estão sendo enviados, cause uma perturbação no sinal de tal forma que pequenas partes da informação sejam modificadas. Assim, para que o atacante tenha sucesso, ele depende altamente da força da modulação em amplitude, que é o processo pelo qual se adequam os sinais (CURRAN; MILLAR; GARVEY, 2012). Uma solução para evitar este tipo de ataque é a checagem da integridade dos dados ao receber as mensagens.

## 3.6 Aplicações

Diversas são as aplicações projetadas para uso do NFC. Pode-se enumerar duas categorias quanto à finalidade do uso do NFC pela aplicação: Identificação e Compartilhamento de Informações (YAQUB; SHAIKH, 2012).

Aplicações de Identificação podem ser utilizadas em uma série de ações cotidianas, como ao chegar no trabalho e bater o ponto ou ir a universidade e se identificar para entrar. Pagar a passagem no metrô ou mesmo seu almoço com apenas um “toque” soa atrativo e inovador. O NFC renova a forma como as pessoas interagem com o ambiente ao seu redor.

Aplicações de Compartilhamento de Informações soam ainda mais atrativas. Imagine chegar no ponto de ônibus e ver um cartaz de um novo filme a estrear. Ao aproximar seu *smartphone* do cartaz, rapidamente você é direcionado para o *trailer* do filme. Ou comprar aquele novo aparelho de som e aproximar seu *smartphone* para baixar o manual do mesmo.



Figura 7 – NFC e suas diversas aplicações.

Fonte: NFC Forum (2014). Traduzida pelo autor.

O NFC é capaz de introduzir novos hábitos às pessoas, fazendo-as pensar como poderiam viver sem a tecnologia.

## Pagamento digital

O NFC foi pensado para que seu *smartphone* substitua seus cartões de crédito. Ele permite que seu *smartphone* funcione como um cartão inteligente, compatível com as tecnologias de cartões inteligentes já existentes no mercado. Tecnologias para pagamento digital como Visa<sup>®</sup> PayWave<sup>™</sup> e MasterCard<sup>®</sup> PayPass<sup>™</sup> são compatíveis com NFC, inclusive já sendo utilizadas em diversos países como Reino Unido, Eslováquia e Brasil (ROEBUCK, 2012). Em 2012, nas Olimpíadas de Londres, Visa e Samsung foram decisivas para a divulgação do NFC, pois muitos dos estabelecimentos e serviços aceitavam como forma de pagamento a tecnologia Visa<sup>®</sup> PayWave<sup>™</sup>, a qual diversos aparelhos produzidos pela Samsung eram compatíveis por possuir *hardware* NFC (CLARK, 2012).

## Transportes

Com NFC, é possível pagar os *tickets* de ônibus e metrô somente encostando seu *smartphone* na catraca. Em 2008, a operadora de trens alemã Deutsche Bahn lançou um programa piloto onde 200 viajantes utilizaram seus *smartphones* com NFC em uma *tag*

no início e fim do percurso da viagem. Assim, eles puderem calcular o valor para cada passageiro e adicionar na cobrança mensal do serviço (AGRAWAL; BHURARIA, 2012). Mais recentemente no Brasil, foi realizado um programa piloto para transporte público com base em pagamento utilizando *tickets* NFC. Diversos ônibus e metrô estavam aptos a receber pagamentos utilizando *smartphones* equipados com NFC (GEMALTO, 2013).

## Saúde

NFC expande a forma como cuidar de pacientes é realizada. Pacientes utilizando *tags* podem ser identificados mais facilmente, além de facilitar para o médico o registro de informações como remédios e horários de aplicação ou diagnósticos. Além disso, pode funcionar como relatório, onde se é possível ver qual foi a hora da última visita do médico a este paciente, ou quais enfermeiras realizaram procedimentos (AGRAWAL; BHURARIA, 2012).

Alguns hospitais e cidades ao redor do mundo já utilizam tecnologias RFID com este fim, como foi o caso do sistema criado no Paquistão para tratar e monitorar crianças diagnosticadas com pneumonia (YAQUB; SHAIKH, 2012). O Serviço Nacional de Saúde do Reino Unido conduz um programa piloto chamado O2 Homecare, que consiste no atendimento domiciliar de pacientes. Os colaboradores do projeto podem baixar informações sobre o paciente em *tags* NFC instaladas na casa do paciente e assim dar continuidade ao tratamento. Além disso, as *tags* servem para registro e relatório de condutas e observações (CLARK, 2009).

## 4 A Plataforma Android

Neste capítulo estudaremos o sistema operacional Android<sup>TM</sup>, sistema que vem revolucionando o mercado de dispositivos móveis. Veremos uma introdução ao sistema, quem o desenvolve, seu propósito e aspectos importantes de sua arquitetura.

Este capítulo também permite um maior entendimento desta monografia, visto que a solução apresentada foi desenvolvida utilizando o sistema operacional Android<sup>TM</sup>. Assim, podemos ambientar as tecnologias que foram utilizadas na construção do Priceshare.

### 4.1 Introdução

O aumento da quantidade de dispositivos móveis que pudemos observar nos últimos anos tem impacto direto no cotidiano das pessoas. A cada dia, essas pessoas convivem com um universo de tecnologias que as ajudam desde tarefas simples às mais complexas. Hoje, o mercado de *smartphones* é um dos setores de *hardware* mais lucrativo e popular. Essa popularização tem gerado cada vez mais novos *softwares* e recursos que fazem uso desse tipo de dispositivo.

A cada nova tecnologia que é incorporada a esses dispositivos, desenvolvedores são estimulados a criar novas aplicações que se utilizam desta. Faz-se então necessário uma plataforma moderna o suficiente para que os desenvolvedores possam acompanhar essas evoluções e, ao mesmo tempo, fazer o bom uso das tecnologias já existentes. Essa plataforma, que é o sistema operacional do dispositivo, deve ser também simples e atraente, para que os usuários se sintam confortáveis e seguros para poder executar as aplicações sem receio.

Uma solução utilizada para isso é o sistema operacional Android<sup>TM</sup>. O Android<sup>TM</sup> é uma plataforma para o desenvolvimento de aplicações móveis, que conta com diversas aplicações já instaladas e com um ambiente de desenvolvimento poderoso e flexível (LECHETA, 2013).

O Android<sup>TM</sup> é um projeto de iniciativa do grupo Open Handset Alliance<sup>TM</sup> (OHA), liderado pela Google<sup>®</sup> e que conta com mais de 80 membros atuantes no desenvolvimento do Android<sup>TM</sup>. Dentre esses membros, cabe listar grandes nomes como HTC, LG, Samsung, Sony, Intel, ASUS, etc. Tem por objetivo definir um sistema aberto para dispositivos móveis de forma a atender as necessidades do consumidor final. Ainda, pretende fornecer uma plataforma flexível e poderosa para o desenvolvimento de aplicações móveis (Open Handset Alliance, 2014).

Com base em sua proposta, o Android<sup>TM</sup> apresenta uma forma inovadora de utilizar

as novas tecnologias, tornando-as mais acessíveis ao usuário. Seu visual moderno e agradável faz com que os usuários se sintam bem ao navegar entre as diversas telas e aplicações que já vem nativas com o Android<sup>TM</sup>. Sua forma de interagir com o usuário, promovendo uma melhor interação fazem com que ele não se sinta perdido e tenha total domínio sobre as ações realizadas.

Sendo assim, desenvolver aplicações para o Android<sup>TM</sup> é simples, uma vez que traz ambiente de desenvolvimento e linguagem de programação comuns aos desenvolvedores. Com o uso da linguagem Java e de ambientes de desenvolvimento já bem conhecidos como o Eclipse, muitos desenvolvedores se sentem confortáveis ao criar aplicações com o Android SDK (acrônimo para *Software Development Kit*).

O Android<sup>TM</sup> é baseado em outro sistema operacional livre, o Linux. Isso permite sua estabilidade e confiabilidade, uma vez que o Linux é um sistema com anos de desenvolvimento e melhorias. Os aplicativos são executados tal qual a arquitetura do gerenciamento de processos do Linux, o que facilita ainda mais por ser uma arquitetura bem conhecida. Cada aplicação é um processo no sistema, e cada processo possui uma *thread* dedicada. Ainda, quando uma nova aplicação é instalada, um usuário é criado para ter acesso somente a estrutura de diretórios da aplicação. Assim, nenhuma outra aplicação interfere diretamente no funcionamento da outra (LECHETA, 2013).

As aplicações desenvolvidas em Java rodam em cima de uma máquina virtual. No Android<sup>TM</sup>, a máquina virtual otimizada para o sistema se chama Dalvik. Quando um desenvolvedor compila sua aplicação, ele gera um *bytecode* que é convertido para formato DEX (*Dalvik Executable*) e empacotado com outros recursos, como imagens e vídeos, em um único arquivo chamado APK (*Android Package File*), que representa o arquivo final para distribuição.

O Android<sup>TM</sup> é um sistema operacional completamente livre e de código aberto (*open source*). Isso é uma vantagem do ponto de vista onde mais desenvolvedores podem contribuir com o sistema, melhorando-o e corrigindo falhas. Ainda, é possível que os fabricantes possam personalizar o sistema operacional Android<sup>TM</sup> de forma a utilizar apenas os recursos providos pelo seu dispositivo. O preço do produto final também sai mais barato, uma vez que não é preciso pagar por uma licença para utilizar o sistema.

Mais de 70% do *smartphones* lançados e já existentes executam o Android<sup>TM</sup> (GARTNER, 2013). O Android<sup>TM</sup> possibilita que as tecnologias embarcadas nestes dispositivos sejam facilmente exploradas ao prover ferramentas intuitivas aos desenvolvedores. Com isso, o Android<sup>TM</sup> vem contribuindo com essa evolução dos dispositivos móveis.

## 4.2 Componentes e Tecnologias

O padrão de programação Android<sup>TM</sup> define certos componentes para que se construa uma aplicação utilizando determinados conceitos para arquitetura do projeto. Uma das arquiteturas de construção seguidas é o padrão *Model-view-controller* (MVC), que divide os componentes do *software* em três camadas: modelos, visualizações e controladores.

Estudaremos a seguir os principais componentes e tecnologias utilizados na construção do Priceshare, para o melhor entendimento do próximo capítulo.

### 4.2.1 View

*View* compreende à camada de visualização do modelo MVC. É o componente utilizado na construção de elementos visuais para interação com o usuário. Assim, ele poderá compreender melhor a informação de forma clara e objetiva.

A construção de boas interfaces garante o sucesso da aplicação. Isso se dá pois o usuário, ao utilizar uma aplicação, deseja entender de forma rápida e fácil o que ele pode fazer e como deve proceder para realizar uma determinada ação.

No Android<sup>TM</sup>, a construção de telas é feita utilizando arquivos XML (*eXtensible Markup Language*). Uma série de etiquetas representam os mais diversos componentes, como exemplo botões e listas. O desenvolvedor tem total controle sobre diversas características dos elementos gráficos, podendo alterar cores, tamanho, fonte, etc.

Há também gerenciadores de *layout*, utilizados na organização dos elementos gráficos em diversas telas. Hoje, no mercado, há diversos dispositivos móveis com diferentes tamanhos de tela. Assim, esses gerenciadores otimizam o espaço onde será renderizado os componentes e permitem que a aplicação sempre mantenha a disposição dos componentes em qualquer resolução de tela utilizada.

### 4.2.2 Activity

*Activity* é a classe que representa a camada de controle do modelo MVC. Ela controla o estado da aplicação, faz a conexão entre modelos e outras telas e define o comportamento de cada componente gráfico da interface, descrito pela *View*.

A *Activity* é uma das classes mais importantes do Android<sup>TM</sup>, pois ela promove a modularidade dos aplicativos e permite o compartilhamento de funcionalidades. *Activities* possuem ciclo de vida próprio, controlados pelo sistema operacional (MEDNIEKS et al., 2013).

A compreensão do ciclo de vida, apresentado na Figura 8, é de extrema importância para entender a arquitetura de desenvolvimento do Android<sup>TM</sup>. Apesar de o Android<sup>TM</sup>

controlar o ciclo de vida de uma *Activity*, é papel do desenvolvedor também se preocupar com os estados que está assume.

Desta forma, quando se está executando uma *Activity* e uma outra aplicação, como por exemplo, uma ligação, interrompe sua execução, esta *Activity* passa a estar em segundo plano. Nesse momento, o Android™ define um estado que representa a situação dessa *Activity*. A partir daí, o usuário pode querer voltar a executar essa *Activity* ou simplesmente não utiliza-la mais. Em ambos os casos, deve se definir o que fazer com as informações para que possam ser recuperadas posteriormente e qual comportamento a *Activity* deve realizar com base na decisão do usuário. Assim, o desenvolvedor precisa conhecer todo o ciclo de vida de uma *Activity* e tomar as devidas ações prevendo os possíveis estados.

Uma *Activity* possui uma série de métodos utilizados também para comunicação com a camada de visualização. Desse modo, é possível alterar em tempo de execução determinadas características dos componentes gráficos ou mesmo iniciar uma outra *Activity*.

### 4.2.3 Intent

*Intent* representam ações a serem executadas pelo Android™. Assim, quando uma aplicação deseja realizar uma tarefa do sistema operacional, ele envia uma mensagem na forma de um *Intent*. Cabe ao sistema operacional interpretar essa mensagem e realizar a ação, que pode ser, como exemplo, efetuar uma ligação ou abrir uma outra aplicação (LECHETA, 2013).

Um *Intent* é necessário para realizar as mais diversas funcionalidades do Android™. Há três casos fundamentais para utilização de um *Intent* (ANDROID, 2014):

- a) Iniciar uma *Activity*: Neste caso, deve ser informado qual *Activity* será iniciada. O *Intent* pode conter quaisquer dados necessários para a execução da nova *Activity*. Pode-se também esperar por resultados após a finalização da *Activity* iniciada;
- b) Iniciar um Serviço: Um Serviço (*Service*) é um componente que realiza operações em segundo plano no sistema operacional, sem interação com o usuário;
- c) Enviar uma Mensagem para todas as Aplicações: Ao enviar uma mensagem para todas as aplicações, caberá a estas saber lidar com a mensagem recebida. Geralmente, este tipo de *Intent* é utilizado com eventos do próprio sistema, como por exemplo ao iniciar o carregamento da bateria do dispositivo.

### 4.2.4 Handler

*Handler* é uma classe que permite o envio ou agendamento de mensagens para serem executadas depois de um determinado intervalo de tempo. Assim, é possível passar informações que são executadas em uma *thread* para outra diferente (LECHETA, 2013).

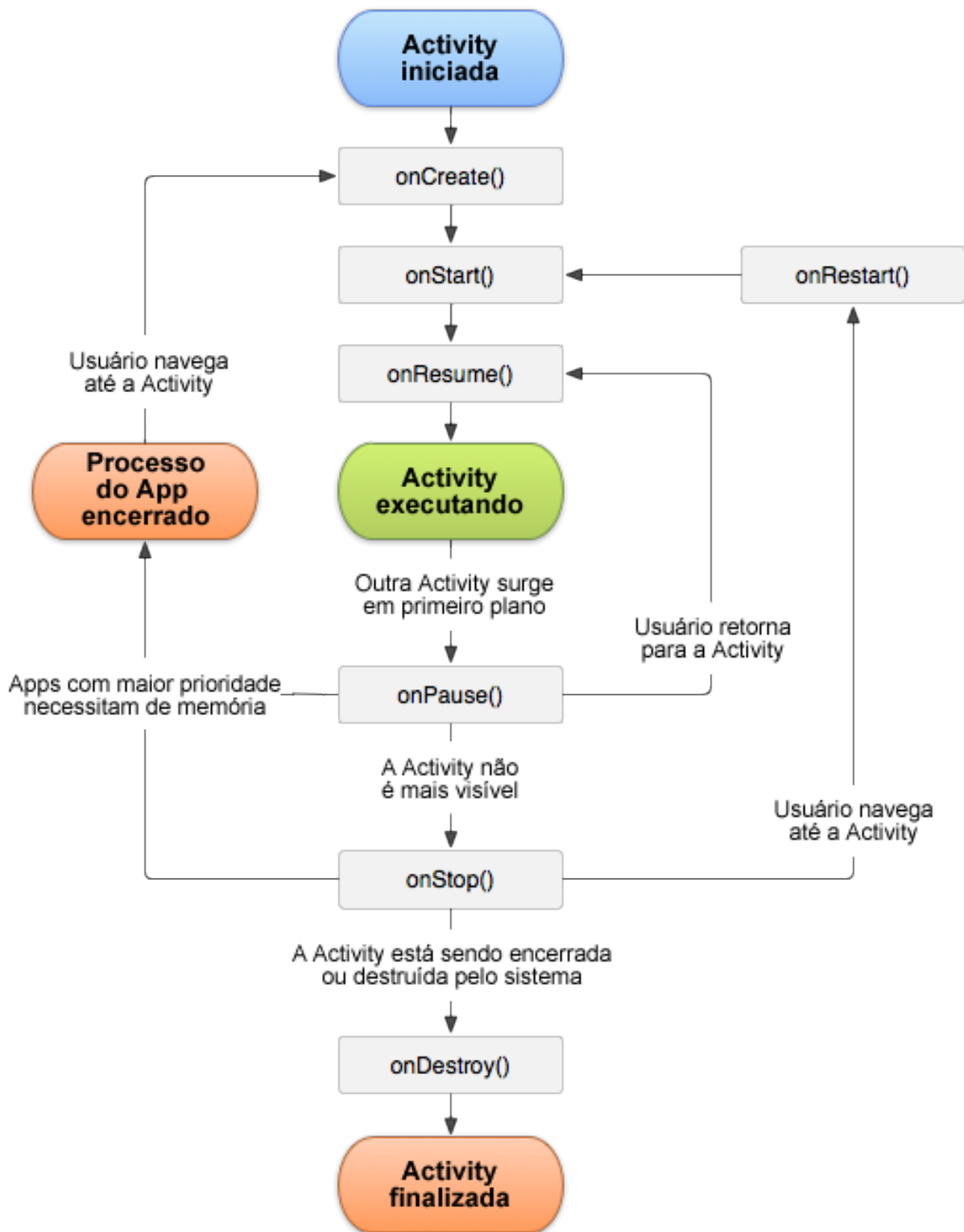


Figura 8 – Ciclo de Vida de uma Activity no Android™.

Fonte: (ANDROID, 2014). Traduzido pelo autor.



Há dois principais usos para um Handler:

- a) Agendar mensagens ou *threads* para serem executadas em algum outro momento;
- b) Enfileirar uma ação para ser realizada em uma outra *thread* diferente da atual.

Quando um processo é criado pela aplicação, a *thread* principal é dedicada para executar uma fila de mensagens, responsável por receber e gerenciar todos os objetos da aplicação e quaisquer telas criadas. Novas *threads* criadas podem se comunicar novamente com a *thread* principal através de um Handler. Isto é realizado através de chamadas a métodos específicos para que as mensagens sejam agendadas na fila de mensagens e processadas quando apropriado. Exemplos para o uso de Handlers são quando existe a necessidade de atualização da interface gráfica de uma *thread* diferente da principal ou quando se recebe informações de sensores e recursos do dispositivo móvel, como NFC (ANDROID, 2014).

#### 4.2.5 NFC e Android

O sistema operacional Android<sup>TM</sup> suporta a tecnologia NFC desde sua versão API (do acrônimo em inglês *Application Programming Interface*) nível 9. Porém, na API nível 14, foi introduzido uma forma mais fácil de trocar mensagens NFC chamada Android Beam.

Atualmente, o Android<sup>TM</sup> suporta todos os modos básicos de operação do NFC. Além disso, o desenvolvimento da biblioteca para suporte do NFC no Android<sup>TM</sup> segue a especificação técnica disponível no NFC Forum. O formato NDEF é utilizado para armazenar dados em *tags* e no compartilhamento de informações entre dispositivos.

Ao ler dados NDEF em uma *tag*, o sistema para tratamento de *tags* analisa os dados descobertos e categoriza-os apropriadamente. Para que uma aplicação manipule os dados de uma *tag* NFC lida, ela precisa declarar um filtro *Intent* relacionado à categoria dos dados recuperados, para só então poder trabalhar com eles (ANDROID, 2014).

No Android<sup>TM</sup>, existem classes específicas que representam cada estrutura de dados definida pelo formato NDEF. Os tipos conhecidos de aplicações NFC também estão descritos na API do Android<sup>TM</sup>. Assim, o usuário consegue tratar facilmente as *tags* de formatos já especificados. É possível ainda a criação de novos tipos de dados utilizando o formato NDEF. Para isso, o desenvolvedor deverá se preocupar em especificar seu próprio formato de dados, provendo métodos de leitura e escrita para seu formato na *tag* (MEDNIEKS et al., 2013).

O suporte ao modo de operação *Peer-to-peer* foi introduzido na versão da API nível 10. Entre as API nível 10 e 13, eram utilizados métodos de envio em primeiro plano,

que se tornaram obsoletos a partir do nível 14 da API, quando foi introduzido o Android Beam.

O Android Beam permite a um dispositivo enviar mensagens NDEF para outro dispositivo apenas aproximando-os. Para que isso ocorra, os dispositivos devem estar com suas telas destravadas e o dispositivo que deseja iniciar a transmissão deve ter a *Activity* que suporta o compartilhamento de dados utilizando NFC em primeiro plano.

O tipo de interação promovido pelo Android Beam é bem mais fácil e intuitivo que o utilizado por outras tecnologias de comunicação sem fio, como Bluetooth<sup>®</sup>, pois não há necessidade de emparelhamento ou configurações. A conexão é automaticamente iniciada quando os dispositivos estão dentro do alcance do *hardware* NFC. Diversas aplicações podem fazer uso dessa tecnologia para compartilhar contatos, páginas web, vídeos para outros dispositivos, etc (ANDROID, 2014).

## 5 Priceshare: um software para o problema da Cotação e Comparação de Preços

Neste capítulo, apresenta-se o Priceshare, um *software* para dispositivos móveis de sistema operacional Android<sup>TM</sup>, que realiza a cotação e comparação de preços, além de permitir o compartilhamento dessas informações entre os usuários.

O objetivo da construção do Priceshare é apresentar a tecnologia *Near Field Communication* e exemplificar como ela pode ser utilizada em nosso cotidiano. Ao mesmo tempo, demonstra como uma aplicação colaborativa se beneficia com o uso do NFC.

### 5.1 Arquitetura do Sistema

O sistema foi idealizado num contexto onde um grupo de pessoas, interessadas em cotar os preços de um determinado conjunto de produtos comuns entre eles, criam o que chamamos de lista de desejo. Esta lista de desejo é compartilhada entre todos, onde cada um fica responsável em cotar os preços em um local diferente dos demais. Após todos concluírem sua tarefa, eles novamente se encontram para trocarem as listas entre si ou para descarregá-las em um computador. Assim, com as listas em um mesmo dispositivo, o *software* se encarrega de compará-las e gerar uma nova lista, que apresenta o menor preço encontrado para cada produto. Para realizar todo este trabalho, é necessário ainda alguém para coordenar todas as ações, como designar locais diferentes para as pessoas envolvidas e evitar possíveis problemas que possam surgir.

O *software* Priceshare foi desenvolvido com base num cenário reduzido, criado para esta monografia e representado na Figura 9. Nele, duas ou mais pessoas se reúnem com objetivo de criar uma lista de desejo comum entre eles. Com todos de posse da mesma lista de desejo, eles então decidem os locais onde cada um deve ir para cotar os preços dos produtos. Terminadas as cotações, eles se reúnem novamente para trocar suas listas de desejo. Com todos de posse das listas de desejo dos demais, eles poderão então comparar as listas e obter uma nova lista, com os produtos de menor preço e seus respectivos locais.

Sendo assim, o cenário reduzido utilizado pelo Priceshare necessita de três itens para que seus usuários possam agir colaborativamente:

- a) Regras, para que todos possam agir de maneira coordenada. O contexto do *software* define as regras a serem seguidas;

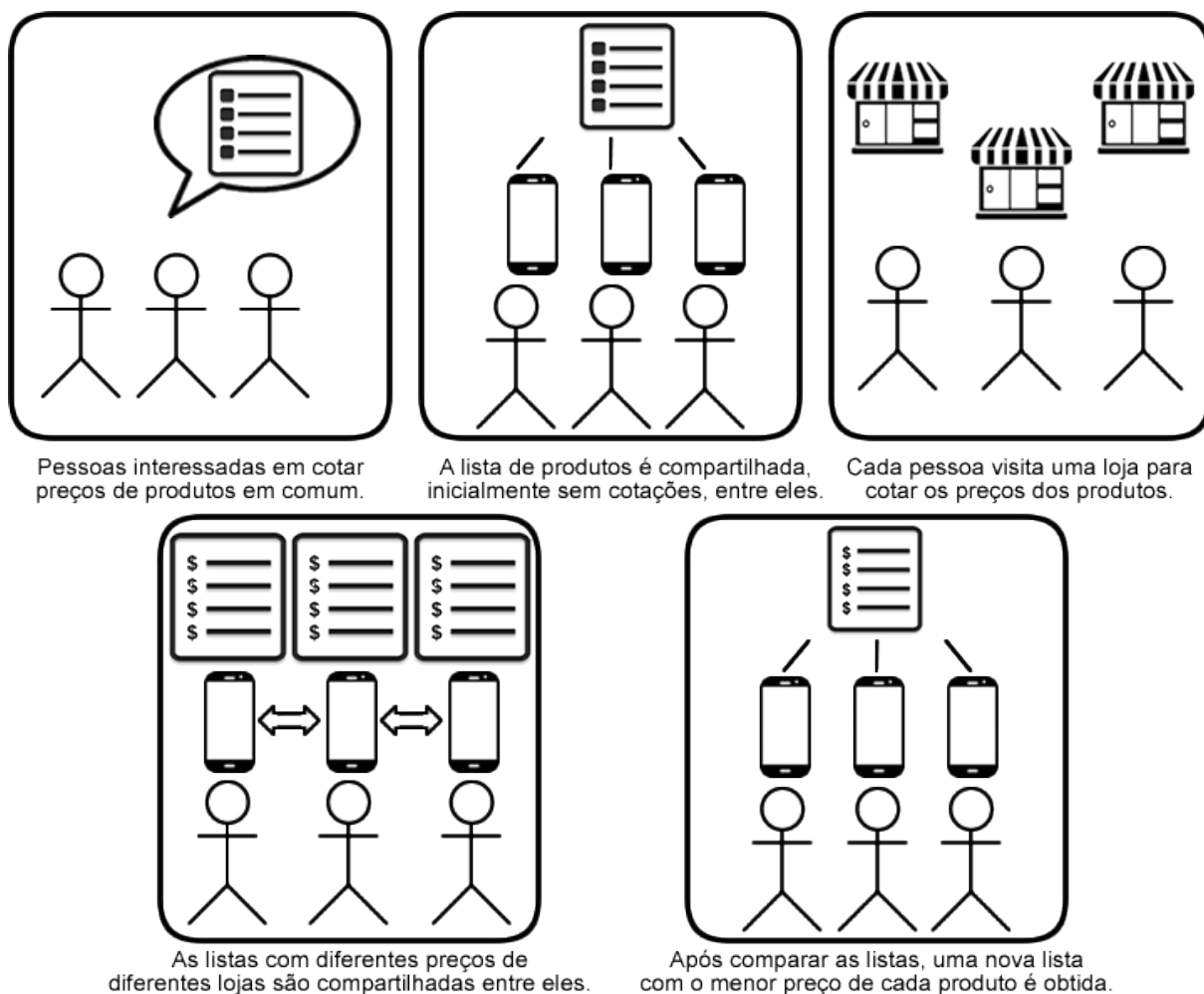
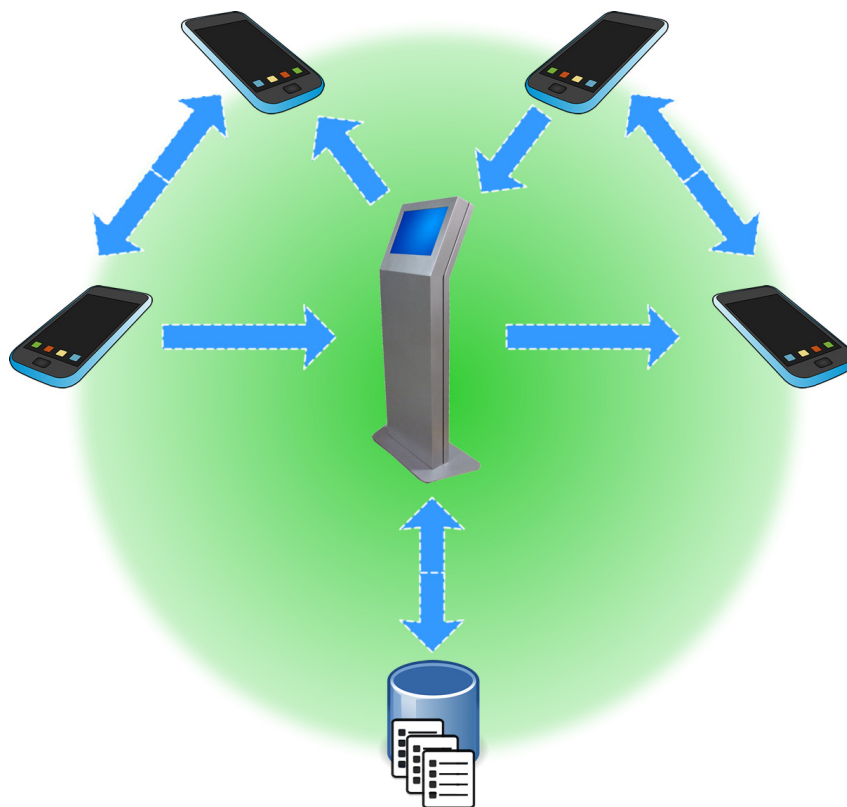


Figura 9 – Cenário Reduzido onde o Priceshare foi idealizado.

- b) Comunicação, para que possam reconhecer os produtos desejados em comum e para trocarem o resultado de suas tarefas;
- c) Cooperação, para que todos concluam sua tarefa com êxito e no tempo estipulado.

O Priceshare insere-se na definição de cada um dos itens, mas exerce papel essencial na cooperação, pois o espaço compartilhado entre seus usuários se torna a lista de desejo. Todos eles são responsáveis por preencher os preços dos itens da lista e de compartilhar essas informações a todos os demais.

A arquitetura do *software* contém diversos elementos que interagem formando um ecossistema favorável à colaboração. O componente principal é a lista de desejos. O aplicativo para dispositivo móvel baseado em Android<sup>TM</sup> tem por finalidade a criação, edição e compartilhamento de listas entre usuários da aplicação. Além desses elementos, há um computador, geralmente em forma de *totem*, para o descarregamento de listas a fim de alimentar um repositório *online*. Este repositório armazenará diversas listas para que possam ser acessadas, localmente ou através da *internet*, as informações descarregadas por

Figura 10 – Arquitetura do *software* Priceshare.

outros usuários. O esquema descrito pode ser visualizado na Figura 10. Nesta monografia, não serão contemplados todos os elementos desta arquitetura. O repositório e a aplicação que será executada nos *totens* deverão ser implementados em trabalho futuros.

## 5.2 Modelagem do Software

Com o objetivo de facilitar a cooperação e a comunicação entre as pessoas, o *software* tem por base a troca de listas de desejo. Uma lista de desejo possui nome, descrição e data de criação. Possui ainda produtos com informações específicas para cada lista, como preço e quantidade. A quantidade de um produto em uma lista é um valor relacionado com a unidade de medida padrão para o produto.

Um produto possui nome, unidade de medida padrão e está associado obrigatoriamente a uma categoria. Pode ainda ser associado a uma subcategoria. A unidade de medida padrão representa, para um produto, a grandeza em que se pode medi-lo ou quantificá-lo.

Os valores para a unidade de medida padrão de um produto, utilizados para o protótipo desta monografia, são baseados nas medidas para volume, massa ou unidade. Assim, para volume utilizamos o litro e para massa utilizamos o quilograma. Quando um produto pode ser medido apenas por unidade, como por exemplo, copos descartáveis, utilizamos a medida genérica unidade.

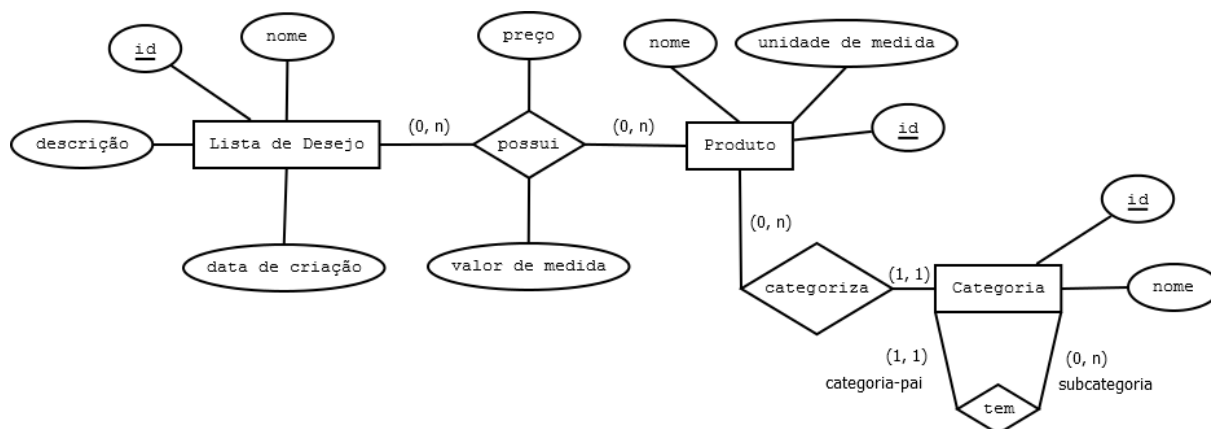


Figura 11 – Modelo conceitual dos dados da aplicação Priceshare.

Uma categoria possui um nome e existe para classificar grupos de produtos relacionados ou análogos. Uma categoria pode ter nenhuma ou várias subcategorias atreladas. Uma subcategoria é uma especialização de uma categoria. Categorias e subcategorias são formas de melhor relacionar os produtos. Elas também auxiliam no processo de distinção de produtos que possuem nomes iguais quando importados de listas de outros usuários, permitindo a comparação de preços entre os produtos. Este processo é explicado com detalhes na próxima seção deste capítulo.

Um produto inserido em uma lista de desejo possui um preço associado a ele naquela lista. Esse preço significará o valor do produto cotado em um determinado local. O preço e a quantidade são utilizados para calcular o valor real do produto a fim de viabilizar a comparação com as demais cotações. Por exemplo, considere o produto “Arroz” cadastrado em duas listas A e B. Na lista A, possui preço igual a R\$ 5,00 e quantidade do produto (medido em quilogramas) igual a 5 Kg. Na lista B, possui preço igual a R\$ 2,00 e quantidade do produto igual 2,5 Kg. Para calcular o valor real do produto, utilizamos a relação preço por quantidade do produto. Assim, o produto “Arroz” se encontra mais barato na lista B, uma vez que seu valor real na lista A é igual a R\$ 1,00 e na lista B é de R\$ 0,80.

Com a descrição do modelo de negócio, foi possível elaborar o modelo conceitual expressado no diagrama de entidade-relacionamento para os dados da aplicação Priceshare, apresentado na Figura 11. O diagrama de entidade-relacionamento foi criado utilizando a ferramenta Dia<sup>1</sup>, um editor para diversos tipos de diagramas.

Analisando o diagrama de entidade-relacionamento, pode se perceber todos os dados descritos. Podemos ainda observar que o atributo “unidade de medida”, da entidade “Produto”, não foi modelado como uma entidade. Este atributo pode apresentar diversos valores a medida que a aplicação é utilizada em outros contextos. Porém, para o cenário

<sup>1</sup> Disponível em <<http://dia-installer.de/>>

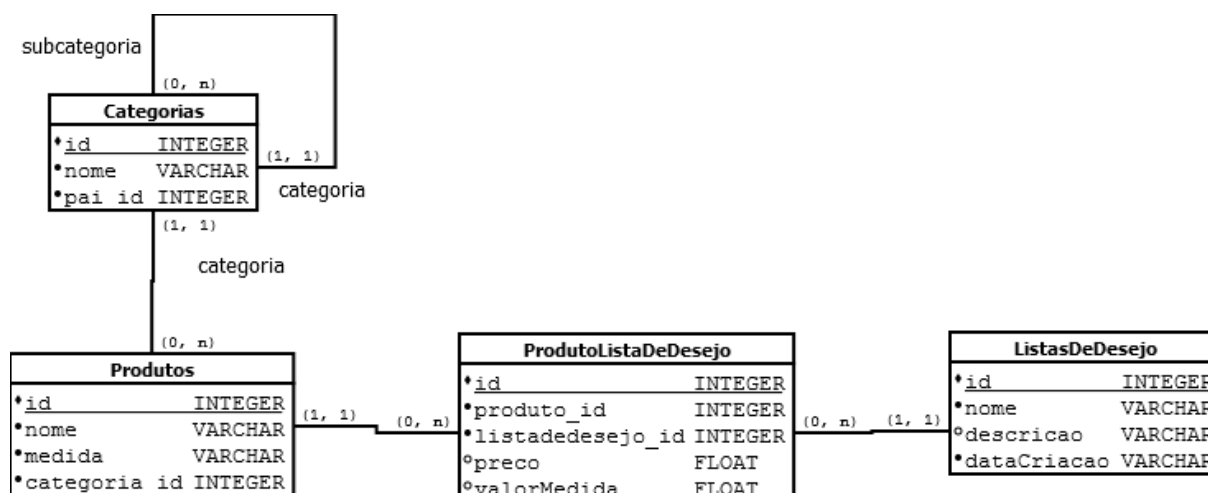


Figura 12 – Modelo lógico gerado a partir do modelo conceitual do Priceshare.

#### Código 1 – Modelo Físico em linguagem SQL do esquema para SQLite.

```

1 CREATE TABLE 'categories' ('parent_id' INTEGER , 'name' VARCHAR NOT NULL
, 'id' INTEGER PRIMARY KEY AUTOINCREMENT );
2 CREATE TABLE 'products' ('category_id' INTEGER NOT NULL , 'measure'
VARCHAR NOT NULL , 'name' VARCHAR NOT NULL , 'id' INTEGER PRIMARY KEY
AUTOINCREMENT , UNIQUE ('name'));
3 CREATE TABLE 'productwishlist' ('wishlist_id' INTEGER NOT NULL , 'user'
VARCHAR , 'product_id' INTEGER NOT NULL , 'price' FLOAT , '
measureValue' FLOAT , 'id' INTEGER PRIMARY KEY AUTOINCREMENT );
4 CREATE TABLE 'wishlists' ('createdDate' VARCHAR NOT NULL , 'description'
VARCHAR , 'user' VARCHAR , 'name' VARCHAR NOT NULL , 'id' INTEGER
PRIMARY KEY AUTOINCREMENT );
5 CREATE INDEX 'categories_name_idx' ON 'categories' ( 'name' );
6 CREATE INDEX 'products_name_idx' ON 'products' ( 'name' );
7 CREATE INDEX 'wishlists_name_idx' ON 'wishlists' ( 'name' );

```

restrito desta monografia, não haveriam muitos valores diferentes para este atributo, validando-se assim esta opção de projeto.

A partir do modelo conceitual, se pode criar o modelo lógico, representado na Figura 12. A partir dele, precisou-se definir o SGBD (Sistema de Gerenciamento de Banco de Dados) a ser utilizado, para que se pudesse então expressar o modelo físico. Foram observados os SGBDs SQLite<sup>2</sup> e H2 Database<sup>3</sup> em suas versões para Android<sup>TM</sup>. Ambos possuem desempenho razoável, embora alguns desenvolvedores Android<sup>TM</sup> apontem que H2 Database é levemente mais otimizado que SQLite. Com um dispositivo com desempenho médio de memória e processamento, e com um volume de dados relativamente pequeno, não há diferença significativa entre os dois SGBDs. Porém, optou-se em utilizar o SQLite, uma vez que ele possui bibliotecas nativas no SDK do Android<sup>TM</sup>. Assim, pudemos construir o modelo físico, apresentado no Código 1, em linguagem SQL para SQLite.

<sup>2</sup> Disponível em <<http://www.sqlite.org>>

<sup>3</sup> Disponível em <<http://www.h2database.com>>

Tabela 3 – Afinidade de Tipos de Dados para o SQLite

Tipos de Dados	Tipo de Dado Afim
INT, INTEGER, TINYINT, SMALLINT, MEDIUMINT, BIGINT, UNSIGNED BIG INT, INT2, INT8	INTEGER
CHARACTER(20), VARCHAR(255), VARYING CHARACTER(255), NCHAR(55), NATIVE CHARACTER(70), NVARCHAR(100), TEXT, CLOB	TEXT
BLOB, tipos de dados não especificados	NONE
REAL, DOUBLE, DOUBLE PRECISION, FLOAT	REAL
NUMERIC, DECIMAL(10,5), BOOLEAN, DATE, DATETIME	NUMERIC

Fonte: Disponível em <<http://www.sqlite.org/datatype3.html>>. Traduzido pelo autor.

No SQLite, não temos alguns tipos de dados encontrados nos SGBDs mais populares, como PostgreSQL<sup>4</sup> e MySQL<sup>5</sup>. Logo, os tipos de dados de alguns atributos das entidades foram traduzidos segundo a tabela de afinidade de tipos de dados para SQLite, conforme verificado na Tabela 3. Outros atributos, como “dataCriacao” da tabela “ListasDeDesejo” que representa uma data, não foram mapeados corretamente seguindo a afinidade de tipos de dados definida pelo SQLite. Isso se deve ao fato da utilização de uma biblioteca para mapeamento objeto-relacional na fase de desenvolvimento do *software* e que será abordada mais profundamente na próxima seção.

Para expressar as funcionalidades do Priceshare, utilizou-se alguns diagramas especificados em UML (do acrônimo em inglês *Unified Modeling Language*) 2.0, como o diagrama de casos de uso e o diagrama de sequência. Os diagramas especificados em UML 2.0 permitem visualizar um sistema e auxiliar no projeto e construção do mesmo. Todos os diagramas utilizados nesta monografia foram criados utilizando o *software* Astah<sup>6</sup>, na versão Community.

O diagrama de casos de uso ilustrado na Figura 13 exemplifica o que um usuário pode realizar com o Priceshare. Ao utilizar o termo “Gerenciar” em alguns casos de uso especificados, a intenção é citar as operações CRUD (do acrônimo em inglês *Create, Retrieve, Update and Delete*) realizadas sobre os dados básicos do sistema.

A funcionalidade que permite executar a operação de troca de lista está exemplificada no caso de uso “Trocar Lista de Desejo com outro usuário”. É nesta funcionalidade que utilizamos o NFC para a troca de listas de desejo. Quando um dispositivo que estiver

<sup>4</sup> Disponível em <<http://www.postgresql.org/>>

<sup>5</sup> Disponível em <<http://www.mysql.com/>>

<sup>6</sup> Disponível em <<http://astah.net/>>



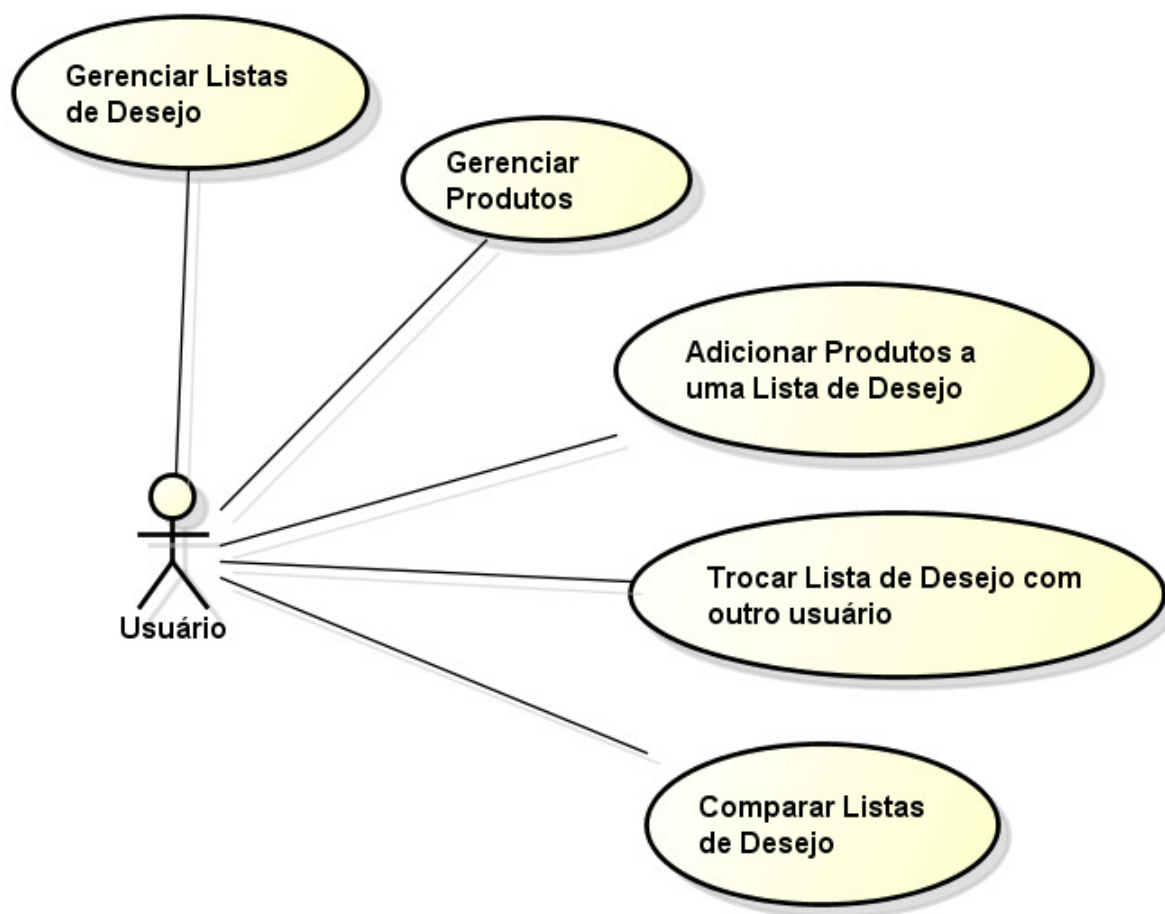


Figura 13 – Diagrama de Caso de Uso para um usuário do Priceshare.

na tela de exibição de uma lista de desejo se aproxima de outro dispositivo, é disparado um evento NFC e a troca de dados é realizada. O outro dispositivo recebe então essa lista e a grava em sua própria base de dados. Isso pode ser observado no diagrama de sequência “Trocar Lista NFC”, ilustrado na Figura 14.

O caso de uso “Comparar Listas de Desejo” representa o cenário onde o usuário, após obter a lista de desejo de outro usuário, as compara e obtém uma nova lista de desejo. Essa nova lista de desejo contém apenas os produtos que apresentam os menores preços entre as duas listas anteriores. Caso haja produtos não comuns entre as duas listas comparadas, estes produtos serão adicionados na nova lista.

O diagrama de classes, apresentado na Figura 15, é resultado do mapeamento do modelo físico, que é um modelo relacional, para um modelo orientado a objetos. O mapeamento objeto-relacional foi importante para satisfazer o paradigma de orientação a objetos e tornar fácil a implementação do *software*.

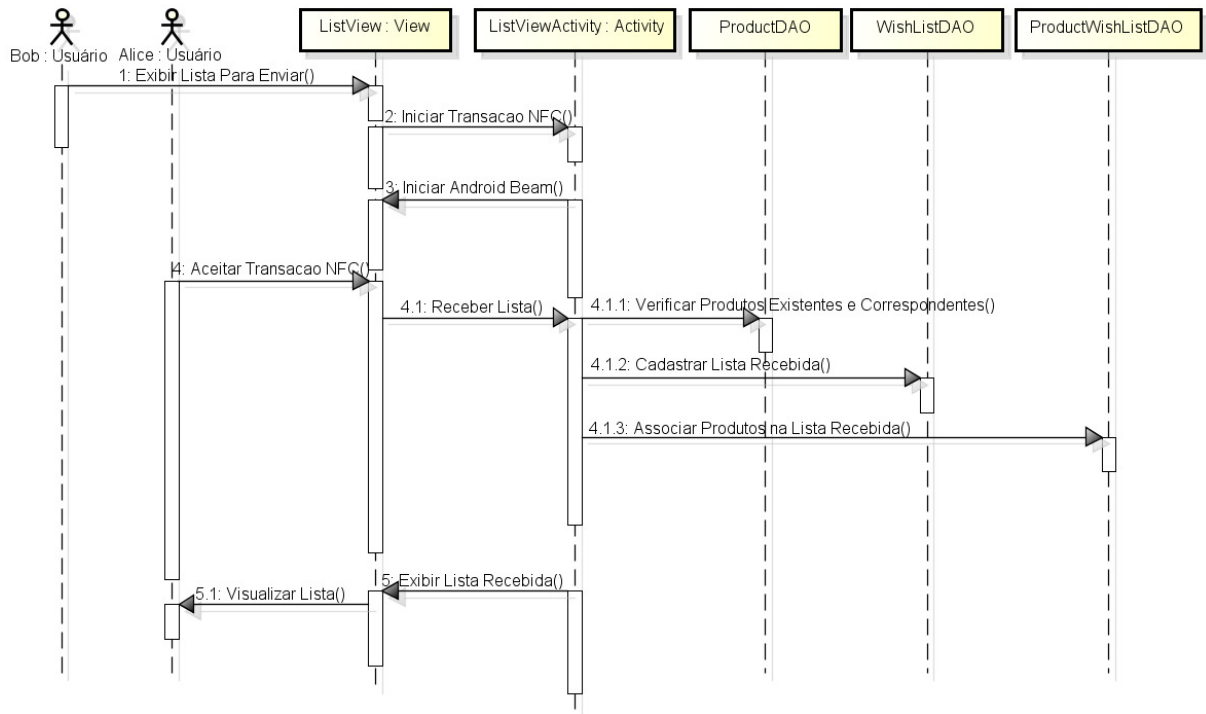


Figura 14 – Diagrama de Sequência para ilustrar o caso de uso “Trocar Lista de Desejo com outro usuário”.

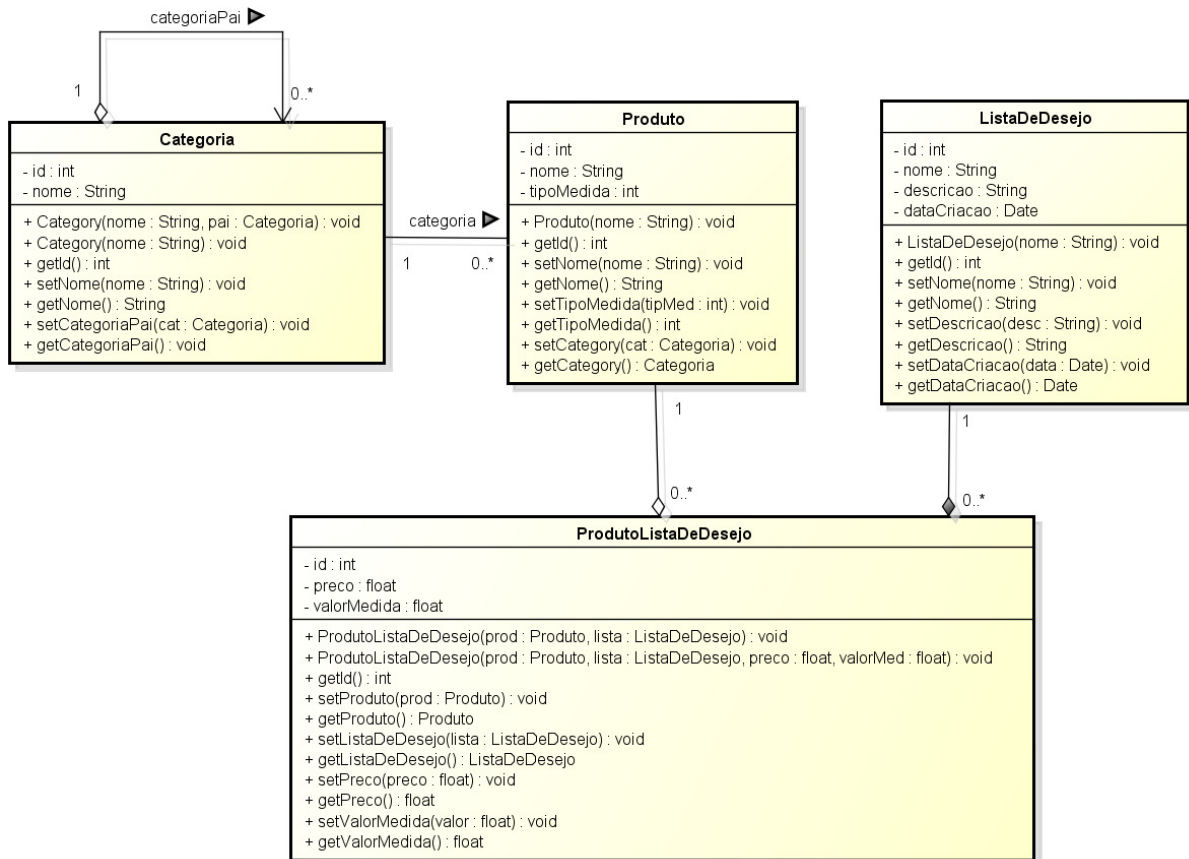


Figura 15 – Diagrama de Classes do Priceshare.

### 5.3 Aspectos de Implementação do Software

A escolha do sistema operacional Android<sup>TM</sup> se deve ao fato de que mais de 70% dos smartphones vendidos utilizam este sistema (GARTNER, 2013). Além disso, possui uma documentação de fácil acesso e entendimento. O Android SDK é livre e possui um pacote já configurado com IDE (do inglês *Integrated Development Environment*), máquina virtual e bibliotecas integradas.

A aplicação foi desenvolvida usando o ambiente integrado de desenvolvimento para Android<sup>TM</sup>, o Eclipse ADT<sup>7</sup> (do inglês *Android Developer Tools*). O Eclipse ADT é um *plugin* para o *software* Eclipse<sup>8</sup>, que amplia as capacidades da ferramenta com recursos para construção de interface gráfica e opções de depuração.

O Priceshare foi implementado utilizando a linguagem de programação Java. Essa escolha facilitou ao se trazer todos os conceitos modelados nos diagramas para a implementação. Foram utilizados outros conceitos como *Model-View-Controller* e Mapeamento Objeto-Relacional (em inglês *Object-relational mapping*, ORM).

MVC é um modelo de arquitetura de *software* que o divide em três camadas. Seu objetivo é garantir a reusabilidade de códigos e a distinção de partes da aplicação, melhorando a interdependência entre elas. O modelo (*model*) é a representação dos dados da aplicação e o acesso a eles. A visualização (*view*) é a apresentação destes dados de forma que faça sentido para o usuário. O controlador (*controller*) é a camada que intermedia as outras duas camadas (modelo e visualização), definindo o comportamento da aplicação de acordo com a interação do usuário, controlando e mapeando as ações.

Mapeamento objeto-relacional é uma técnica de programação para minimizar as dificuldades de conversão de dados registrados em banco de dados relacionais para programação baseada em paradigmas de orientação a objetos, e vice-versa. Isso facilita no trabalho de persistência dos dados, uma vez que o mapeamento dos modelos garante o armazenamento e carregamento consistente dos objetos.

Foi utilizada uma biblioteca para o mapeamento objeto-relacional dos modelos do Priceshare. A ORMLite<sup>9</sup> é uma biblioteca Java para persistência dos objetos em bancos de dados SQL. Possui uma versão para o sistema operacional Android<sup>TM</sup> com uso do SQLite, SGBD nativo do Android<sup>TM</sup>. Dentre suas funcionalidades, vale citar o uso de anotações Java para facilitar o mapeamento dos dados e uma poderosa camada para abstração de classes para objetos de acesso a dados (DAO, do acrônimo em inglês *Data Access Object*), permitindo a separação das regras de negócios e das regras de acesso ao banco de dados.

Após a implementação das classes com base nos modelos, as anotações Java

<sup>7</sup> Disponível em <<http://developer.android.com/tools/sdk/eclipse-adt.html>>

<sup>8</sup> Disponível em <<https://www.eclipse.org/>>

<sup>9</sup> Disponível em <<http://ormlite.com/>>.

Código 2 – Trecho da classe Produto com anotações Java para uso com ORMLite

```
1 @DatabaseTable(tableName = "products")
2 public class Product
3 {
4     @DatabaseField(generatedId = true)
5     private int id; // Identificador
6
7     @DatabaseField(index = true, unique=true, canBeNull=false)
8     private String name; // Nome do Produto
9
10    @DatabaseField(foreign = true, foreignAutoRefresh=true, canBeNull=
11        false)
12    private Category category;
```

fornecidas pela ORMLite foram inseridas para permitir a criação e atualização das tabelas da base de dados SQLite. A primeira vez que um objeto da classe é instanciado, o esquema da tabela é criado ou atualizado a partir das rotinas internas do ORMLite. Isso é feito com o uso da classe `OrmLiteSqliteOpenHelper`. Ela funciona como um *helper* para recuperação do DAO de cada modelo do Priceshare, além de criar a conexão com a base de dados.

Para que os objetos pudessem ser trocados entre as *Activities*, foi necessário estender as classes do modelo utilizando a interface `Parcelable`. `Parcelable` é uma interface cujas instâncias podem ser escritas e restauradas de um objeto `Parcel`. Um objeto `Parcel` é um contêiner para uma mensagem que contém referências a dados e objetos. É utilizado como mecanismo de alto desempenho para transporte de dados entre processos. Uma vez que cada *Activity* é um processo e as *Activities* do Priceshare trocam objetos entre si por meio de seus controladores, a solução se mostrou eficiente (ANDROID, 2014).

Quando dois usuários desejam trocar listas de desejo no Priceshare, eles utilizam NFC. Uma transação NFC é iniciada quando um dos dispositivos, visualizando uma lista de desejo, aproxima do outro dispositivo. Neste momento, a lista que está sendo visualizada é transferida para o outro dispositivo e um sistema de importação se encarrega de manter os dados sincronizados entre os dois.

Os objetos são enviados através da rede através de uma técnica chamada serialização de dados, disponível em Java com o uso da interface `Serializable`. Serialização é o processo de armazenamento ou transporte de dados em *bytes* ou formato de texto para que se possa recuperar um objeto a seu estado original. Em Android™, `Parcelable` só garante o transporte entre processos que estejam operando em um mesmo dispositivo, uma vez que é dependente de *hardware* e versão de sistema operacional. É uma alternativa rápida, mas que não garante o estado original do objeto, por isso não sendo recomendado para transporte em rede (ANDROID, 2014; ORACLE, 2014).

A troca de listas de desejos é feito por um recurso chamado Android Beam. Este recurso foi criado com a finalidade de permitir o rápido compartilhamento entre urls para

Código 3 – Trecho de código para inicialização do NFC usando Android Beam.

```
1 // Recupero o adaptador NFC
2 nfcAdapter = NfcAdapter.getDefaultAdapter( this );
3
4 if ( nfcAdapter != null )
5 {
6     // Associo as callbacks utilizadas pelo Android Beam
7     nfcAdapter.setNdefPushMessageCallback( new CreateNdefMessage(
8         getApplicationContext(), prodWls.toArray() ), this );
9     nfcAdapter.setOnNdefPushCompleteCallback( new OnNdefPushComplete(
10        getApplicationContext() ), this );
11 }
```

*sites da web*, contatos e outros dados através do NFC. Em essência, o que o Android Beam faz é utilizar o modo de operação *Peer-to-peer* do NFC para possibilitar a troca dos dados entre os dispositivos Android<sup>TM</sup>.

O trecho de código exibido em Código 3 demonstra a inicialização do adaptador NFC e a associação de duas *callbacks* (métodos disparados na realização de determinado evento) que são utilizadas para a criação da mensagem NDEF com um lista de produtos e para a ação a ser tomada após o envio da mensagem, respectivamente.

Quando a mensagem é recebida pelo dispositivo receptor, uma *Activity* é executada para cuidar da importação da lista de desejos recebida. Essa importação leva em consideração os produtos recebidos que já existem na base de dados do dispositivo receptor. O esquema da base de dados do Priceshare descreve que um produto possui um atributo nome com valor único. Isso significa que não existem dois produtos com nomes iguais. Assim, faz-se necessário mapear o produto recebido para o produto existente na base de dados.

Um produto é uma entrada definida pelo usuário, como podemos ver na Figura 16 e na Figura 17. Todos os seus dados são inseridos pelo usuário, inclusive o nome. Caso um usuário utilize um nome ligeiramente diferente para um mesmo produto cadastrado na base de dados de outro usuário, se faz necessário realizar um mapeamento entre estes produtos. Como exemplo, os nomes “Sabão” e “Sabão em Barra” podem significar a mesma coisa para usuários com listas de desejo diferentes. Dessa forma, quando eles compararem as listas, sentirão necessidade de que estes produtos também sejam comparados. Este problema não ocorre com usuários com uma mesma lista de desejo.

Para resolução deste problema, não foram utilizadas nenhuma técnica de inteligência artificial ou recursos computacionais, como o uso de etiquetas utilizadas em Web Semântica. Possíveis soluções para problemas similares são abordados por outros autores em outros trabalhos e não fazem parte da proposta desta monografia. Assim, foi desenvolvido uma heurística capaz de resolver parcialmente o problema, com o uso de verossimilhança entre nomes dos produtos e filtragem de atributos em comum como categoria, subcategoria e

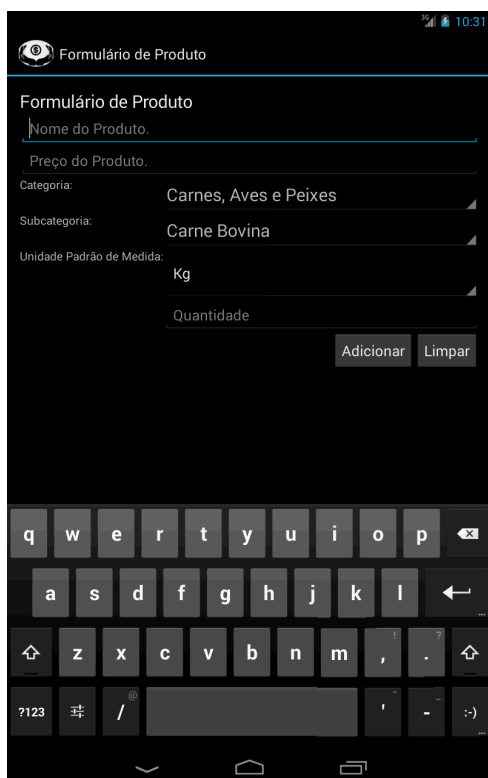


Figura 16 – Tela de cadastro de produtos.

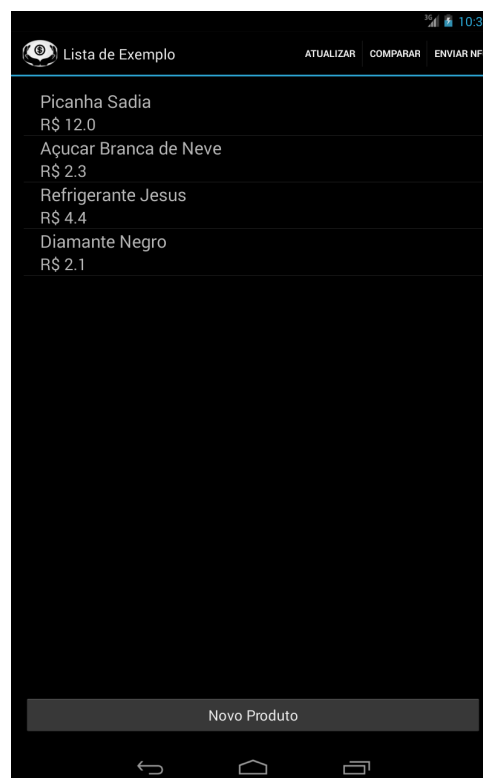


Figura 17 – Exibição de produtos de uma lista.

unidade de medida. O algoritmo desenvolvido e utilizado está expresso na Figura 18.

## 5.4 Testes e Resultados

Foram utilizadas máquinas virtuais do sistema Android<sup>TM</sup> para testes do protótipo Priceshare. As versões do Android<sup>TM</sup> utilizadas nas máquinas virtuais foram 4.2.2 e 4.3. Os emuladores utilizados para criação e manuseio das máquinas virtuais foram o AVD (do acrônimo em inglês *Android Virtual Device*), disponível com a SDK do Android<sup>TM</sup>, e o VirtualBox<sup>10</sup>. A versão 4.3 foi utilizada com o AVD, com uso da imagem Intel x86 fornecida pela Google<sup>®</sup>. A versão 4.2.2 foi utilizada com o VirtualBox, com uso da imagem Intel x86 modificada pela equipe do OpenNFC<sup>TM</sup> para suporte a simulação de NFC.

OpenNFC<sup>TM</sup> é um projeto de código aberto que implementa as funcionalidades NFC com base em um controlador de *chipset* NFC. Além de suportar a maioria das funcionalidades listadas no padrão NFC, há uma camada de abstração de *hardware*, o que garante que a pilha da aplicação sejam independente de *hardware*. Inclui também um *software* para simulação chamado NFC Simulator.

Para testes realizados em ambiente virtual, foi utilizada a imagem criada pelo OpenNFC<sup>TM</sup>. Duas máquinas virtuais, criadas pelo VirtualBox, são ligadas simultanea-

<sup>10</sup> Disponível em <<https://www.virtualbox.org/>>

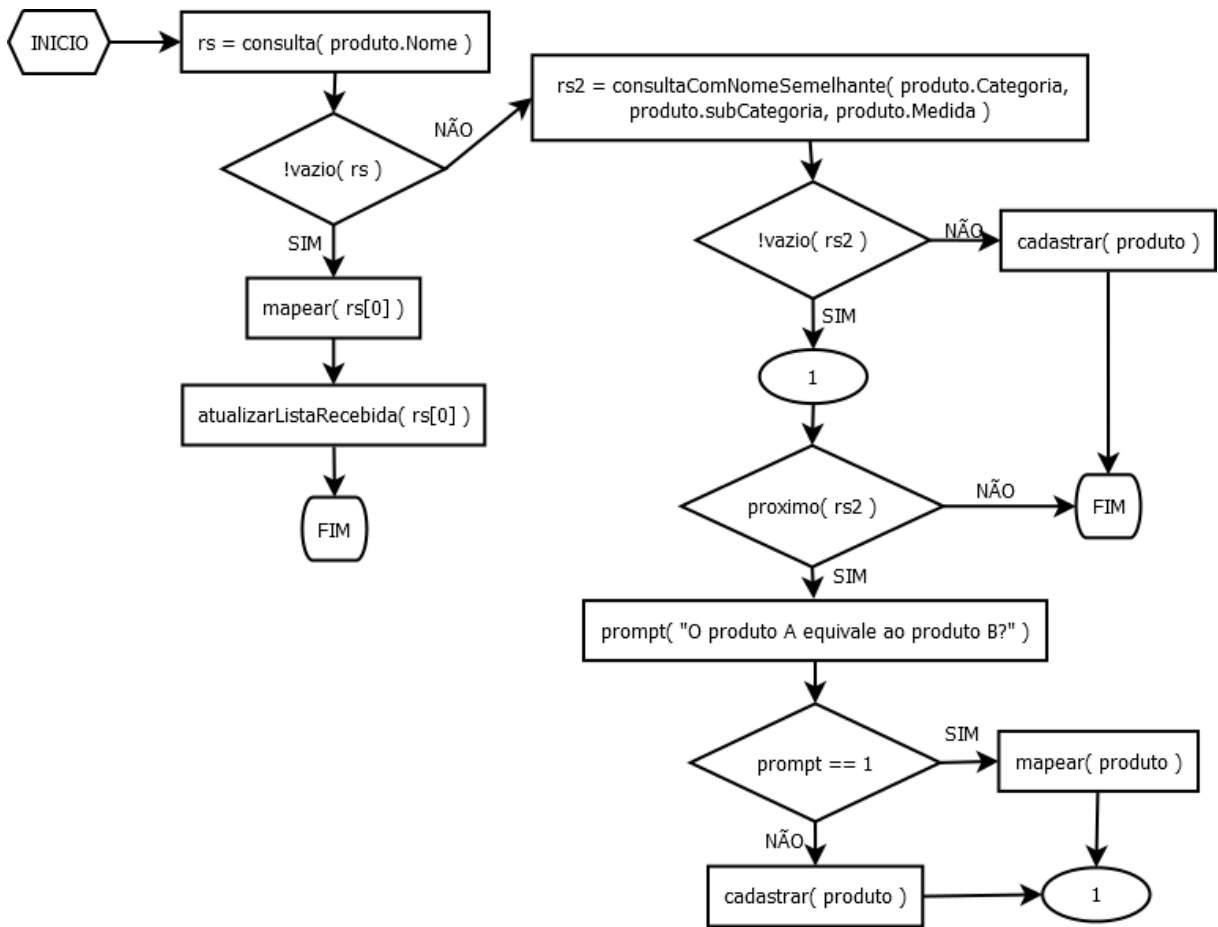


Figura 18 – Fluxograma representando a importação de uma lista recebida de outro usuário.

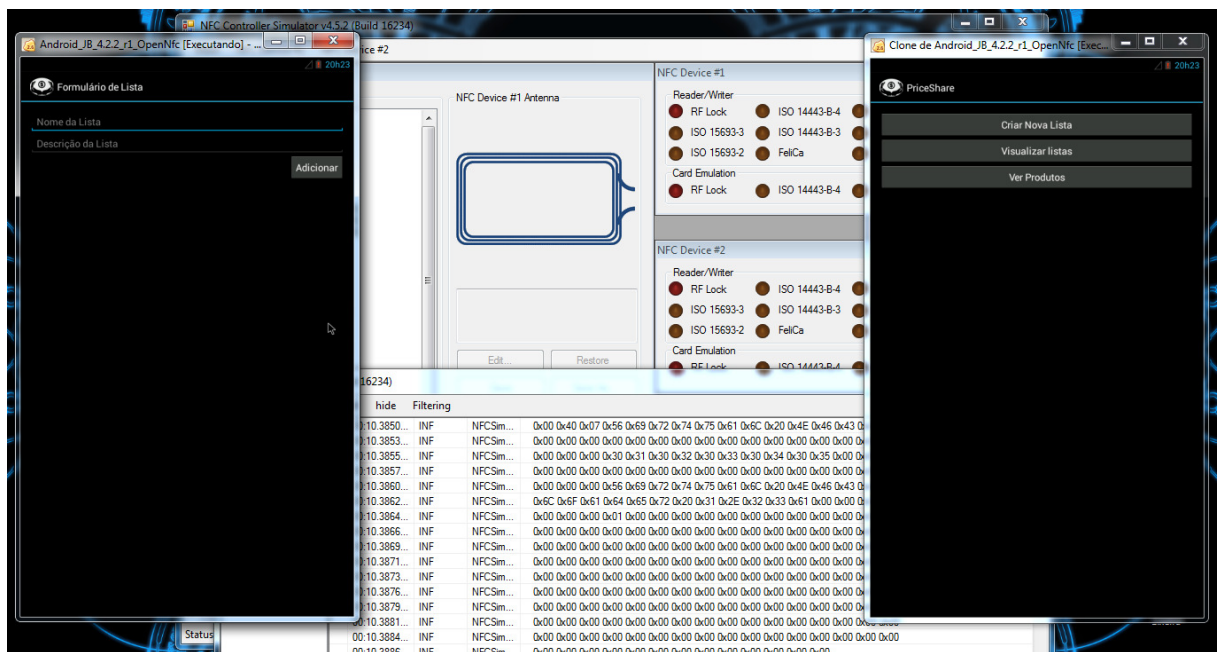


Figura 19 – Ambiente utilizado para testes usando OpenNFC™ e VirtualBox.

mente. O *hardware* NFC é simulado utilizando os aplicativos Connection Center e NFC Simulator, fornecidos no pacote disponível para Microsoft® Windows™ pelo OpenNFC™. Assim que as máquinas virtuais estabelecem a conexão, é possível realizar a depuração das mensagens transmitidas através da aplicação Trace Server, também fornecida pelo OpenNFC™.

A comunicação é iniciada quando em uma das máquinas virtuais o usuário está visualizando os produtos de uma lista de desejo. Assim que a configuração para utilização de um segundo dispositivo NFC é habilitado no NFC Simulator, o dispositivo que deseja enviar a lista entra no modo de operação *Peer-to-peer*, através do Android Beam.

Os testes também foram realizados em dispositivos físicos. Foram utilizados um dispositivo Asus Nexus 7 com a versão 4.3 do Android™ e um dispositivo Samsung Galaxy IV com a versão 4.2.2. Nos dispositivos, o teste se deu de maneira prática e objetiva. Um usuário, ao visualizar os produtos de uma lista de desejo, aproximava seu dispositivo ao dispositivo do outro usuário. Assim, os *hardwares* NFC se detectavam e realizavam a troca dos dados.

Durante ambos os testes com dispositivos físicos e em ambiente virtual, os dados foram trocados de maneira satisfatória. Como os dados transferidos eram listas de desejo com uma quantidade razoável de produtos (foram utilizados no máximo 50 produtos), não foram detectadas perda de dados na transmissões em testes com dispositivos físicos. Um dos problemas que foram visualizados enquanto se desenvolvia o *software* era justamente estimar o tempo necessário em que os dispositivos deveriam ficar em contato para que se realizasse a troca das informações. Nos testes realizados, o tempo de 1 segundo era suficiente para que uma quantidade de 50 produtos pudessem ser enviados sem comprometer a integridade dos mesmos.

Não foi possível desenvolver uma ferramenta para analisar a quantidade de dados e o tempo necessário em que os dispositivos devem permanecer próximos para se garantir a troca de dados. Porém, o gráfico ilustrado na Figura 20 demonstra uma estimativa levando em consideração a velocidade máxima de transferência de 424 Kbits/s entre dispositivos no modo de operação *Peer-to-peer*. Pelo gráfico, podemos perceber que tamanhos de dados maiores que 500 KB levam mais do que 10 segundos para serem transferidos.

Para averiguarmos a satisfação da tecnologia NFC quanto ao seu uso em sistemas colaborativos, tendo como estudo de caso o problema da cotação e comparação de preços, criamos um cenário real para teste. O teste consistia em considerar uma mesma lista de desejo para duas pessoas. Esta lista de desejo continha produtos que facilmente são encontrados em supermercados da cidade de São Luís, Maranhão. De posse da lista de desejo, cada participante deveria cotar os preços em supermercados diferentes. No fim, ambos deveriam trocar novamente as listas e realizar a comparação entre elas através do Priceshare. O resultado deveria ser uma lista de desejo igual para ambos, com os produtos



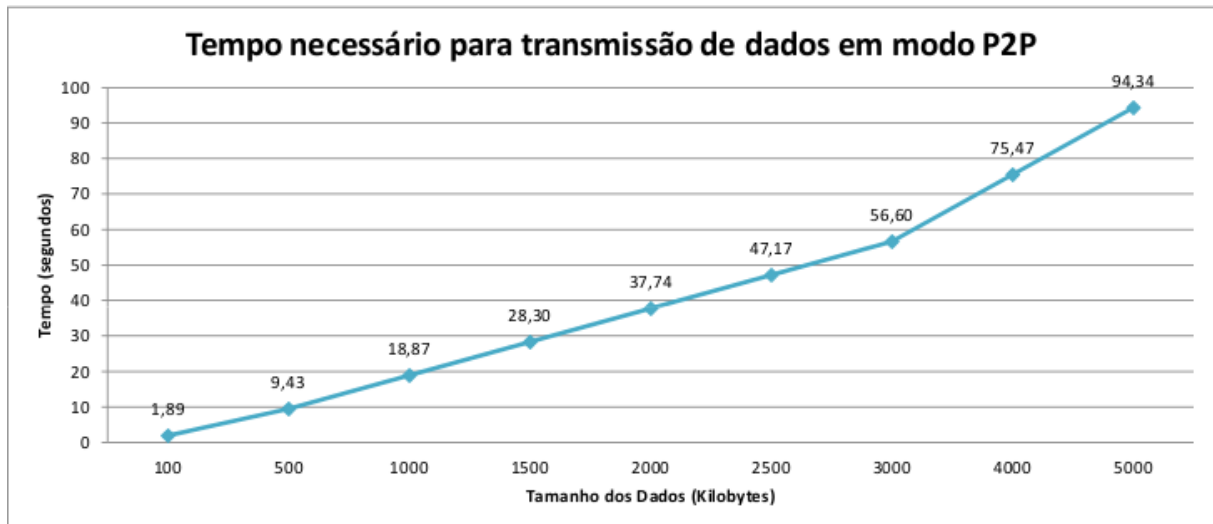


Figura 20 – Análise entre Tamanho dos dados e Tempo durante uma transmissão em Modo *Peer-to-peer*.

de menor preço encontrados em cada supermercado.

Com a realização do teste, pudemos constatar que o resultado obtido foi o mesmo para ambos os participantes. Após a comparação, eles obtiveram uma lista igual. Observou-se que os participantes ficaram satisfeitos com o Priceshare, relatando a sua facilidade ao realizar uma tarefa do cotidiano deles. Isso se deve ao menor esforço para a realização da tarefa, pois ambos precisaram apenas ir uma vez a cada supermercado para concluir onde comprar o produto considerando o preço mais barato.

## 6 Conclusão

Este trabalho apresentou uma solução para o problema de cotação e comparação de preços de forma colaborativa. Assim, foi desenvolvido um protótipo, chamado Priceshare, para validação desta solução. Esta solução dispõe-se do uso de dispositivos móveis e da tecnologia *Near Field Communication*.

O Priceshare foi desenvolvido utilizando o sistema operacional Android<sup>TM</sup>, por ser a plataforma mais utilizada em dispositivos móveis no mundo. Também foi realizado um estudo sobre a tecnologia NFC, analisando-a e expondo seus diversos aspectos. Dessa forma, foi possível conhecer seu potencial e seu uso em diversas aplicações.

O protótipo proposto para o problema de cotação e comparação de preços auxilia no processo de cooperação dos usuários, interessados em comparar preços a partir de uma lista comum de produtos. Com ele, se pôde verificar a eficiência da tecnologia NFC ao ser utilizada em aplicações colaborativas.

Com a utilização do protótipo, chegou-se aos seguintes resultados:

- a) A tecnologia NFC cumpre sua proposta, uma vez que se mantém simples, intuitiva e robusta. Sua arquitetura provê facilidade ao trocar informações, mantendo ainda a segurança e confiabilidade no compartilhamento;
- b) O ambiente de desenvolvimento para Android<sup>TM</sup> é claro e compreensível. A API fornecida para desenvolvimento de aplicações NFC permite explorar os mais diversos recursos da tecnologia;
- c) Aplicações colaborativas beneficiam-se do NFC por ser uma tecnologia presente em diversos dispositivos móveis e de uso mais simples e intuitivo que outras existentes;
- d) A colaboração é uma maneira eficaz de resolver o problema da cotação de preços ao utilizar tecnologias que cooperam para que as pessoas possam pesquisar e comparar preços de produtos, facilitando a troca dessas informações entre os envolvidos.

Diante do exposto, conclui-se que o Priceshare compreende os requisitos necessários de uma solução para o problema da cotação e comparação de preços de forma colaborativa. No entanto, são enumeradas a seguir algumas limitações do protótipo:

- a) O protótipo foi desenvolvido num cenário restrito, onde os produtos a serem cadastrados são em geral do gênero alimentício;

- b) A comunicação entre os usuários para definir a lista de desejo exige a presença e o debate com uso da linguagem oral;
- c) O protótipo não possui suporte à leitura de código de barras ou *tag* RFID específica, tipos de informação registradas e obtidas no Cadastro Nacional de Produtos (CNP);
- d) A comparação de produtos entre listas de desejo diferentes necessitam da intervenção humana para que se possa identificar possíveis produtos iguais expressos com nomes diferentes;
- e) O tamanho da carga de dados influi diretamente no tempo necessário para se manter uma conexão NFC. Listas de desejo com um número elevado de produtos tendem a demorar mais para serem transferidas entre dois dispositivos.

Com essas limitações, sugerem-se algumas melhorias para trabalhos futuros:

- a) Abordar o cenário mais amplo onde o Priceshare foi idealizado, abrangendo o uso de produtos dos mais diversos gêneros e implementação dos demais componentes descritos na arquitetura;
- b) Criar novas formas de comunicação, como envio de mensagens de texto ou áudio, para facilitar a definição de listas de desejo entre os usuários do *software*;
- c) Possibilitar o uso de informações registradas no CNP, facilitando o uso do *software* por parte dos usuários;
- d) Utilizar ou desenvolver um algoritmo com base em inteligência artificial para que seja possível que o *software* possa aprender quais os diferentes nomes que um mesmo produto possa ter e assim identificá-los automaticamente;
- e) Permitir o uso de outras tecnologias de redes sem fio quando o dispositivo móvel não oferecer suporte ao NFC ou quando for necessário transferir uma grande carga de dados.

# Referências

- Acca Software. *Orçamento Tabelas de Preços e Composição - ACCA software*. 2014. Software. Disponível em: <<http://www.accasoftware.com/ptb/id3/primus/index.html>>. Acesso em: 28 jan. 2014. Citado 2 vezes nas páginas 16 e 24.
- AGRAWAL, P.; BHURARIA, S. Near field communication. *IT Matters*, v. 67, 2012. Citado 2 vezes nas páginas 35 e 42.
- ANDROID. *Android API level 19 - References*. 2014. Website. Disponível em: <<http://developer.android.com/reference/packages.html>>. Acesso em: 13 mar. 2014. Citado 5 vezes nas páginas 46, 47, 48, 49 e 59.
- BAKER, N. Zigbee and bluetooth strengths and weaknesses for industrial applications. *Computing Control Engineering Journal*, v. 16, n. 2, p. 20–25, April 2005. ISSN 0956-3385. Citado na página 36.
- BILGINER, B.; LJUNGGREN, P.-L. Near field communication. *Lund University*, 2011. Citado 2 vezes nas páginas 26 e 28.
- Bluetooth SIG. *Bluetooth Technology*. 2014. Website. Disponível em: <<http://www.bluetooth.com>>. Acesso em: 21 jul. 2014. Citado na página 34.
- BORGHOFF, U. M.; SCHLICHTER, J. H. *Computer-supported cooperative work*. [S.l.]: Springer, 2000. Citado na página 20.
- BUSCAPÉ. *Buscapé - Onde você encontra sempre o melhor preço!* 2014. Website. Disponível em: <<http://www.buscape.com.br/>>. Acesso em: 28 jan. 2014. Citado na página 16.
- CHANDLER, N. *What's an NFC tag? - HowStuffWorks.com*. 2014. Website. Disponível em: <<http://electronics.howstuffworks.com/nfc-tag.htm>>. Acesso em: 28 jan. 2014. Citado na página 30.
- CLARK, S. *NHS tests NFC for home care visits*. 2009. News. Disponível em: <<http://www.nfcworld.com/2009/06/05/31279/nhs-tests-nfc-for-home-care-visits/>>. Acesso em: 06 fev. 2014. Citado na página 42.
- CLARK, S. *Visa releases Olympics NFC and contactless usage statistics*. 2012. News. Disponível em: <<http://www.nfcworld.com/2012/09/19/317958/visa-releases-olympics-nfc-and-contactless-usage-statistics/>>. Acesso em: 06 fev. 2014. Citado na página 41.
- CURRAN, K.; MILLAR, A.; GARVEY, C. M. Near field communication. *International Journal of Electrical and Computer Engineering (IJECE)*, v. 2, n. 3, p. 371–382, 2012. Disponível em: <<http://iaesjournal.com/online/index.php/IJECE/article/view/234>>. Citado 3 vezes nas páginas 38, 39 e 40.
- Custo Certo. *Custo Certo - Sistema de Cotações, Comparação de Preços*. 2013. Website. Disponível em: <<http://www.custocerto.com.br/>>. Acesso em: 28 jan. 2014. Citado 2 vezes nas páginas 16 e 24.

- DESAI, E.; SHAJAN, M. G. A review on the operating modes of near field communication. *International Journal of Engineering and Advanced Technology (IJEAT)*, v. 2, n. 2, 2012. ISSN 2249–8958. Disponível em: <<http://www.ijeat.org/attachments/File/v2i2/B0956112212.pdf>>. Citado na página 30.
- Dicto Sistemas. *Rede Cotação - Sistema de cotações, licitações e pedidos online*. 2013. Website. Disponível em: <<https://www.redecotacao.com.br/>>. Acesso em: 28 jan. 2014. Citado 2 vezes nas páginas 16 e 24.
- EBV Elektronik. *RFID Selection Guide*. 1. ed. [S.l.], 2010. Citado 2 vezes nas páginas 27 e 29.
- ELLIS, C. A.; GIBBS, S. J.; REIN, G. Groupware: some issues and experiences. *Communications of the ACM*, ACM, v. 34, n. 1, p. 39–58, jan 1991. Citado 2 vezes nas páginas 19 e 20.
- FUKS, H.; RAPOSO, A. B.; GEROSA, M. A. Engenharia de groupware: Desenvolvimento de aplicações colaborativas. In: *XXI Jornada de Atualização em Informática, Anais do XXII Congresso da Sociedade Brasileira de Computação*. [S.l.: s.n.], 2002. v. 2, p. 89–128. Citado 4 vezes nas páginas 20, 21, 22 e 23.
- GARTNER. *Gartner Says Smartphone Sales Accounted for 55 Percent of Overall Mobile Phone Sales in Third Quarter of 2013*. 2013. News. Disponível em: <<http://www.gartner.com/newsroom/id/2623415>>. Acesso em: 10 fev. 2014. Citado 2 vezes nas páginas 44 e 58.
- GEMALTO. *Rio de Janeiro launches pilot for public transportation NFC ticketing with smartphones*. 2013. News. Disponível em: <[https://www.gemalto.com/php/pr\\_view.php?id=1688](https://www.gemalto.com/php/pr_view.php?id=1688)>. Acesso em: 06 fev. 2014. Citado na página 42.
- HASELSTEINER, E.; BREITFUSS, K. Security in near field communication (nfc). In: *Workshop on RFID Security RFIDSec*. [S.l.: s.n.], 2006. Citado 2 vezes nas páginas 38 e 39.
- IMHONTU, E. E.; KUMAR, O. Y. *A Survey on Near Field Communication in Mobile Phones & PDAs*. 60 p. thesis — Halmstad University, School of Information Science, Computer and Electrical Engineering (IDE), 2010. Citado na página 40.
- Infrared Data Association. *Infrared Data Association*. 2014. Website. Disponível em: <<http://irdajp.info/>>. Acesso em: 04 fev. 2014. Citado 2 vezes nas páginas 34 e 37.
- KUMAR, A. Near field communication. 2011. Citado 2 vezes nas páginas 35 e 39.
- LECHETA, R. R. *Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK*. 3. ed. São Paulo: Novatec Editora, 2013. ISBN 978-85-7522-344-4. Citado 3 vezes nas páginas 43, 44 e 46.
- MEDNIEKS, Z. et al. *Programando o Android*. 2. ed. São Paulo: Novatec Editora, 2013. ISBN 978-85-7522-336-9. Citado 2 vezes nas páginas 45 e 48.
- MEEKER, M. *Mary Meeker Explains Internet 2012 in 17 Minutes: The Full D10 Interview*. 2014. Video. Disponível em: <<http://allthingsd.com/20120612/mary-meeker-explains-internet-2012-in-17-minutes-the-full-d10-interview-video/>>. Acesso em: 28 jan. 2014. Citado na página 17.

- MICHAELIS. Cotação. In: \_\_\_\_\_. *Dicionário Michaelis*. Michaelis, 2014. Disponível em: <<http://www.uol.com.br/michaelis>>. Acesso em: 28 jan. 2014. Citado na página 23.
- MOTA, D. A. R.; CAVALCANTE, A. A. F. Gestão do conhecimento em empresas através de sistemas colaborativos (groupware). 2009. Citado 2 vezes nas páginas 19 e 22.
- NFC Forum. *NFC Data Exchange Format (NDEF)*. 1. ed. [S.l.], 2006. Citado 3 vezes nas páginas 32, 33 e 34.
- NFC Forum. *NFC Forum*. 2014. Website. Disponível em: <<http://nfc-forum.org/>>. Acesso em: 28 jan. 2014. Citado 6 vezes nas páginas 27, 32, 34, 36, 38 e 41.
- NOKIA. *Introduction to NFC*. 1. ed. [S.l.], 2011. Citado na página 32.
- NOKIA. Differences among different nfc tags. In: \_\_\_\_\_. *Nokia Developer Wiki*. Nokia, 2014. Disponível em: <[http://developer.nokia.com/community/wiki/Differences\\_among\\_different\\_NFC\\_tags](http://developer.nokia.com/community/wiki/Differences_among_different_NFC_tags)>. Acesso em: 03 fev. 2014. Citado na página 31.
- NOKIA. Understanding nfc data exchange format (ndef) messages. In: \_\_\_\_\_. *Nokia Developer Wiki*. Nokia, 2014. Disponível em: <[http://developer.nokia.com/community/wiki/Understanding\\_NFC\\_Data\\_Exchange\\_Format\\_%28NDEF%29\\_messages](http://developer.nokia.com/community/wiki/Understanding_NFC_Data_Exchange_Format_%28NDEF%29_messages)>. Acesso em: 03 fev. 2014. Citado na página 32.
- Open Handset Alliance. *Open Handset Alliance*. 2014. Website. Disponível em: <<http://www.openhandsetalliance.com/>>. Acesso em: 02 abr. 2014. Citado na página 43.
- ORACLE. *Java™ Platform, Standard Edition 7 API Specification*. 2014. Website. Disponível em: <<http://docs.oracle.com/javase/7/docs/api/overview-summary.html>>. Acesso em: 13 mar. 2014. Citado na página 59.
- PIMENTEL, M.; FUKS, H. *Sistemas Colaborativos*. [S.l.]: Elsevier Brasil, 2012. ISBN 978-85-352-4669-8. Citado 2 vezes nas páginas 19 e 20.
- ROEBUCK, K. *Near Field Communication (NFC): High-impact Strategies-What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. [S.l.]: Emereo Pub., 2012. Citado na página 41.
- SALOMON, G.; GLOBERSON, T. When teams do not function the way they ought to. *International journal of Educational research*, Elsevier, v. 13, n. 1, p. 89–99, 1989. Citado na página 22.
- WANT, R. Near field communication. *Pervasive Computing, IEEE*, v. 10, n. 3, p. 4–7, Julho 2011. ISSN 1536-1268. Citado na página 28.
- Wi-Fi Alliance. *Wi-Fi Alliance*. 2014. Website. Disponível em: <<http://www.wi-fi.org/>>. Acesso em: 21 jul. 2014. Citado na página 34.
- WIKIPEDIA. Radio-frequency identification. In: \_\_\_\_\_. *Wikipedia, The Free Encyclopedia*. Wikimedia, 2014. Disponível em: <[http://en.wikipedia.org/wiki/Radio-frequency\\_identification](http://en.wikipedia.org/wiki/Radio-frequency_identification)>. Acesso em: 28 jan. 2014. Citado na página 26.
- YAQUB, M. U.; SHAIKH, U. A. Near field communication. 2012. Citado 3 vezes nas páginas 28, 40 e 42.

ZigBee Alliance. *ZigBee Alliance*. 2014. Website. Disponível em: <<http://www.zigbee.org/>>. Acesso em: 04 fev. 2014. Citado 2 vezes nas páginas 34 e 36.