

**EDUARDO GIROTO GARCIA**

**COMPAREOFFER: APLICATIVO ANDROID E SOFTWARE WEB  
PARA OBTENÇÃO E COMPARAÇÃO DE PREÇOS PARA  
PRODUTOS DE SUPERMERCADOS**

Assis,

2012

**EDUARDO GIROTO GARCIA**

**COMPAREOFFER: APLICATIVO ANDROID E SOFTWARE WEB PARA  
OBTENÇÃO E COMPARAÇÃO DE PREÇOS PARA PRODUTOS DE  
SUPERMERCADOS**

Trabalho de conclusão de Curso  
apresentado ao Instituto Municipal de  
Ensino Superior de Assis, como requisito  
do Curso Superior de Ciência da  
Computação

**Orientador:** Me. Douglas Sanches da Cunha

**Área de Concentração:** Informática.

Assis,  
2012

## FICHA CATALOGRÁFICA

G216c	<p data-bbox="379 1205 1436 1388">GARCIA, Eduardo Giroto CompareOffer : Aplicativo Android e Software Web para obtenção e comparação de preços para produtos de supermercado. / Eduardo Giroto Garcia. Assis, 2012. 91p</p> <p data-bbox="379 1422 1436 1534">Orientador: Me. Douglas Sanches da Cunha Trabalho de Conclusão de Curso de Ciência da Computação. Fundação Educacional do Município de Assis – FEMA.</p> <p data-bbox="379 1568 1436 1601">1.Android 2.Java 3.JSF</p> <p data-bbox="1182 1608 1436 1677">CDD: 005.133 Biblioteca / FEMA</p>
-------	---

**EDUARDO GIROTO GARCIA**

**COMPAREOFFER: APLICATIVO ANDROID E SOFTWARE WEB PARA  
OBTENÇÃO E COMPARAÇÃO DE PREÇOS PARA PRODUTOS DE  
SUPERMERCADOS**

Trabalho de Conclusão de Curso apresentado ao  
Instituto Municipal de Ensino Superior de Assis,  
como requisito do Curso Superior de Ciência da  
Computação, analisado pela seguinte comissão  
examinador.

**Orientador:** Me. Douglas Sanches da Cunha

**Analisador:** Me. Fábio Eder Cardoso

Assis,

2012

## DEDICATÓRIA

Dedico este trabalho a Deus, a toda minha família e principalmente aos meus pais Jaime e Ivone que sempre preocuparam em dar-me a melhor educação, e formação do meu caráter, sempre me apoiando, e aos amigos que me ajudaram nessa jornada de trabalho.

## **AGRADECIMENTO**

Agradeço primeiramente a Deus por me dar força para cumprir mais uma etapa da minha vida, dando-me força, saúde e sabedoria, me abençoando nos momentos mais difíceis e por encorajar-me a enfrentar os desafios encontrados ao longo desse caminho.

Agradeço a todos meus familiares, principalmente aos meus pais Jaime e Ivone, pelo grande carinho e incentivo que me deram, quando escolhi fazer este curso, acreditando e dando – me total apoio a minha capacidade e sempre me apoiando em todas as decisões.

Ao meu Orientador Douglas, pela sua orientação, ideias e opiniões, através de seus conhecimentos, não só na área acadêmica, mas também na vida pessoal e estar sempre me incentivando durante este trabalho.

Agradeço também a todos os professores da FEMA com quais tive aula, sem exceções, que diretamente ou indiretamente contribuíram para o meu enriquecimento intelectual.

Agradeço a minha grande amiga Karina, que ao longo desses anos acabou se tornando uma irmã, cultivando uma linda amizade e me ajudando sempre que precisei e aos demais amigos de sala e da Fema; Diego, Stella, Adriel, Erion, Douglas, Elcio, Thiago, Guilherme Mendes, Luis, José Tiago, Eder, Jean, Roberta e Mariana entre outros que estiveram juntos comigo durante esses 4 anos, enfrentando as mesmas dificuldades e ajudando diretamente ou indiretamente a superá-las, sempre com muita alegria e dedicação, e a todos que acreditaram em meu potencial.

Aos amigos do CEPEIN, que conviveram comigo durante 3 anos de estagio, trocando ideias, informações e o mais importante, a amizade.

*Paciência e perseverança têm o efeito mágico de fazer as dificuldades desaparecerem e os obstáculos sumirem. (John Quincy Adams)*

## RESUMO

CompareOffer, é um sistema para auxílio de cotação de preço de produtos de supermercado, dividido em 2 módulos. O Primeiro é um aplicativo mobile para celulares com *Android* (front-end – Aplicação principal), que comunica através de um *WebService* usando o padrão *Restful*, e o outro um software web trabalhando como back-end (Programa Auxiliar). O Sistema *Android* consiste em comparar o preço de um produto em outros estabelecimentos; a partir do momento em que é inserido um código de barras na câmera, o mesmo irá pesquisar se o produto apresenta um preço melhor que o concorrente. Já o sistema Web engloba todos os módulos de cadastro, gerenciamentos e gerenciamentos de ofertas. As tecnologias usadas foram *Java* e *Android* no desenvolvimento dos softwares e o banco de dados o *PostgreSQL*.

**Palavras-chaves:** Java, Android, Restful, PostgreSQL.



## ABSTRACT

CompareOffer, is a computer system to aid price quotation of supermarket products which is divided in 2 modules. The first one is a mobile application for *android* cellphones(front-end – Main application), which communicates with a WebService using the *Restful* architectural style, and the second as a back-end web software(Aid Program).The *android* system consists on comparing prices of a product from another places; for the moment when a code bar is inserted on its camera, it will search whether the product has a better price in another place. The web system encompasses all the entries, managements and offer managements. For the software's development the technologies used were *Java* and *Android* and *PostgreSQL* for Databases.

**Keywords:** Java, Android, Restful, Eclipse, PostgreSQL

## Lista de Figuras

Figura 1: Porcentagem dos tipos de <i>Android</i> utilizados .....	20
Figura 2: Arquitetura <i>Android</i> .....	21
Figura 3: As empresas que compõe a Aliança - Open Handset Alliance .....	24
Figura 4 - Mapa Mental .....	29
Figura 5: Sequenciamento de Atividade.....	31
Figura 6: Cronograma .....	32
Figura 7: Diagrama Entidade Relacionamento (DER) .....	33
Figura 8: Pacote Geral .....	34
Figura 9: Pacote Úti.....	34
Figura 10: Diagrama de Classes .....	35
Figura 11: Pacote DAO .....	36
Figura 12: Diagrama de Casos de Uso .....	38
Figura 13: Diagrama de Sequencia - Manter Oferta.....	39
Figura 14: Manter Oferta - Use Case 1 .....	40
Figura 15: Diagrama de Sequência - Visualizar Histórico .....	41
Figura 16: Visualizar Histórico - <i>Use Case 2</i> .....	42
Figura 17: Diagrama de Sequência - Manter Administrador Supermercado .....	44
Figura 18: Manter Administrador Supermercado - <i>Use Case 3</i> .....	45
Figura 19: Diagrama de Sequência - Manter Rua .....	46
Figura 20: Manter Rua – <i>Use Case 4</i> .....	47
Figura 21: Diagrama de Sequência - Manter Bairro .....	49
Figura 22: Manter Bairro - <i>Use Case 5</i> .....	50
Figura 23: Diagrama de Sequência - Manter Cidade .....	51
Figura 24: Manter Cidade - <i>Use Case 6</i> .....	52
Figura 25: Diagrama de Sequência - Manter Produto .....	
Figura 26: Manter Produto – <i>Use Case 7</i> .....	55
Figura 27: Diagrama de Sequência - Manter Supermercado Conveniado .....	56
Figura 28: Manter Supermercado Conveniado – <i>Use Case 8</i> .....	57

Figura 29: Diagrama de Sequência - Manter Fornecedor .....	59
Figura 30: Manter Fornecedor – Use Case 9 .....	60
Figura 31: Diagrama de Sequência - Emitir Relatório de Supermercado .....	61
Figura 32: Emitir Relatório de Supermercado – Use Case 10.....	62
Figura 33: Diagrama de Sequência - Emitir Relatório de Fornecedor .....	63
Figura 34: Emitir Relatório de Fornecedor – Use Case 11 .....	64
Figura 35: Diagrama de Sequência - Emitir Relatório de Produto.....	65
Figura 36: Emitir Relatório de Produto – Use Case 12.....	66
Figura 37: Diagrama de Sequência - Buscar Produto .....	67
Figura 38: Buscar Produto – Use Case 13 .....	68
Figura 39: Diagrama de Sequência - Realizar Compra .....	69
Figura 40: Buscar Produto – Use Case 14 .....	70
Figura 41: Diagrama de Sequência - Acompanhar Gastos .....	71
Figura 42: Acompanhar Gastos – Use Case 15 .....	71
Figura 43: Diagrama de Sequência - Emitir Relatório Lista de Compra .....	72
Figura 44: Emitir Relatório Lista de Compra – Use Case 16 .....	73
Figura 45: Diagrama de Sequência - Manter Lista de Compra .....	74
Figura 46: Manter Lista de Compra – Use Case 17 .....	75
Figura 47: Diagrama de Atividade - Aplicação <i>Android</i> .....	76
Figura 48: Diagrama de Atividade: Website destinado ao Supermercado .....	77
Figura 49: Diagrama de Atividade – Website destinado ao Administrador.....	78

## Lista de Tabelas

Tabela 1: Especificação de <i>Use Case</i> 1 - Manter Oferta .....	40
Tabela 2: Especificação de <i>Use Case</i> 2 - Visualizar Histórico .....	42
Tabela 3: Especificação de <i>Use Case</i> 3 - Manter Administrador Supermercado .....	45
Tabela 4: Especificação de <i>Use Case</i> 4 - Manter Rua .....	47
Tabela 5: Especificação de <i>Use Case</i> 5 - Manter Bairro .....	50
Tabela 6: Especificação de <i>Use Case</i> 6 - Manter Cidade .....	52
Tabela 7: Especificação de <i>Use Case</i> 7 - Manter Produto .....	55
Tabela 8 - Especificação de <i>Use Case</i> 8 - Manter Supermercado Conveniado .....	57
Tabela 9: Especificação de <i>Use Case</i> 9 - Manter Fornecedor .....	60
Tabela 10: Especificação de <i>Use Case</i> 10 - Emitir Relatório de Supermercado .....	62
Tabela 11: Especificação de <i>Use Case</i> 11 - Emitir Relatório de Fornecedor .....	64
Tabela 12: Especificação de <i>Use Case</i> 12 - Emitir Relatório de Produto .....	66
Tabela 13: Especificação de <i>Use Case</i> 13 - Buscar Produto .....	68
Tabela 14: Especificação de <i>Use Case</i> 14 – Realizar Compra .....	70
Tabela 15: Especificação de <i>Use Case</i> 15 - Acompanhar Gastos .....	72
Tabela 16: Especificação de <i>Use Case</i> 16 - Emitir Relatório Lista de Compra .....	73
Tabela 17: Especificação de <i>Use Case</i> 17 - Manter Lista de Compra .....	75

## SUMÁRIO

1. Introdução.....	16
1.2. Objetivo .....	17
1.3. Justificativa.....	17
1.4. Método de Desenvolvimento.....	17
2. Tecnologias para Desenvolvimento.....	19
2.1. Android.....	19
2.1.1. Plataforma Android .....	21
2.1.2. Android e seus nomes .....	23
2.1.3. Open Handset Alliance (OHA) .....	24
2.2. Java.....	25
2.2.1. Java Serve Faces (JSF) .....	26
2.3. PostgreSQL.....	26
2.4. Metodologias e Arquiteturas .....	27
2.4.1. Casos de Uso .....	27
2.4.2. Diagrama de Classes.....	27
2.4.3. Diagrama de Entidade Relacionamento .....	28
2.4.4. Diagrama de Atividades .....	28
2.4.5. Sequenciamento de atividades .....	28
3. Documentação.....	29
3.1. Mapa Mental .....	29
3.2. WBS.....	30
3.3. Sequenciamento de Atividades.....	31
3.4. Cronograma .....	32
3.5. Diagrama Entidade Relacionamento (DER).....	33
3.6. Diagrama de Classes.....	34
3.6.1. Pacote Geral.....	34
3.6.2. Pacote Útil .....	34

3.6.3.	Pacote Model.....	35
3.6.4.	Pacote DAO.....	36
3.7.	Casos de Uso.....	37
3.7.1.	Especificações do Ator Administrador Supermercado.....	38
3.7.1.1.	Diagrama de Sequência – Manter Oferta.....	39
3.7.1.1.1.	Manter Oferta - Use Case 1 .....	39
3.7.1.2.	Diagrama de Sequência - Visualizar Histórico .....	41
3.7.1.2.1.	Visualizar Histórico - Use Case 2 .....	42
3.7.2.	Especificações do Ator Administrador Sistema .....	43
3.7.2.1.	Diagrama de Sequência - Manter Administrador Supermercado ....	43
3.7.2.1.1.	Manter Administrador Supermercado - Use Case 3.....	44
3.7.2.2.	Diagrama de Sequência – Manter Rua .....	46
3.7.2.2.1.	Manter Rua - Use Case 4 .....	47
3.7.2.3.	Diagrama de Sequência – Manter Bairro .....	48
3.7.2.3.1.	Manter Bairro - Use Case 5.....	49
3.7.2.4.	Diagrama de Sequência – Manter Cidade .....	51
3.7.2.4.1.	Manter Cidade - Use Case 6 .....	52
3.7.2.5.	Diagrama de Sequência - Manter Produto.....	53
3.7.2.5.1.	Manter Produto – Use Case 7 .....	54
3.7.2.6.	Diagrama de Sequência - Manter Supermercado Conveniado .....	56
3.7.2.6.1.	Manter Supermercado Conveniado – Use Case 8 .....	57
3.7.2.7.	Diagrama de Sequência - Manter Fornecedor .....	58
3.7.2.7.1.	Manter Fornecedor - Use Case 9 .....	59
3.7.2.8.	Diagrama de Sequência - Emitir Relatório de Supermercado.....	61
3.7.2.8.1.	Emitir Relatório de Supermercado - Use Case 10.....	62
3.7.2.9.	Diagrama de Sequência - Emitir Relatório de Fornecedor.....	63
3.7.2.9.1.	Emitir Relatório de Fornecedor - Use Case 11 .....	63
3.7.2.10.	Diagrama de Sequência - Emitir Relatório de Produto.....	65
3.7.2.10.1.	Emitir Relatório de Produto – Use Case 12.....	65
3.7.3.	Especificações do usuário Android.....	66

3.7.3.1.	Diagrama de Sequência - Buscar Produto .....	67
3.7.3.1.1.	Buscar Produto - Use Case 13.....	67
3.7.3.2.	Diagrama de Sequência - Realizar Compra.....	68
3.7.3.2.1.	Realizar Compra - Use Case 14.....	69
3.7.3.3.	Diagrama de Sequência - Acompanhar Gastos .....	70
3.7.3.3.1.	Acompanhar Gastos – Use Case 15 .....	71
3.7.3.4.	Diagrama de Sequência - Emitir Relatório Lista de Compra .....	72
3.7.3.4.1.	Emitir Relatório Lista de Compra - Use Case 16.....	73
3.7.3.5.	Diagrama de Sequência - Manter Lista de Compra .....	74
3.7.3.5.1.	Manter Lista de Compra – Use Case 17 .....	74
3.8.	Diagramas de Atividades .....	76
3.8.1.	Diagrama de Atividade da Aplicação Android.....	76
3.8.2.	Diagrama de Atividade do Website referente ao Supermercado .....	77
3.8.3.	Diagrama de Atividade referente ao Administrador .....	78
4.	Conclusão.....	79
	Referências Bibliográficas .....	80

## 1. Introdução

Cada vez mais pessoas ao redor do mundo estão usando celulares, smartphones e tablets com o sistema operacional *Android*.

O *Android* está se tornando um sistema operacional Mobile de fácil manuseio tanto para os usuários, como para os programadores, e com isso, vemos a cada dia surgirem uma leva de *apps* (programas) para ele, sendo que até Janeiro de 2012 havia no *Play Store* (repositório de aplicativos *Android*) 400 mil aplicativos cadastrados.

É um software que pertence à empresa © *Google*, mas totalmente livre para desenvolvimento de aplicativos que podem ser distribuídos gratuitamente ou pago. Desta forma, boa parte dos aparelhos possuem um hardware que exploram totalmente os recursos do *Android* como wireless, Bluetooth, GPS e os 7 sensores : acelerômetro, orientação, gravidade, giroscópio, luz, pressão e temperatura, todos eles estão acessíveis para desenvolvimento de *apps*, o que faz com que a gama de possibilidades e interações de aplicativos seja grande.

O sistema a ser desenvolvido chama-se *CompareOffer* e terá 2 módulos, o primeiro é um programa baseado na arquitetura *Android* que atenderá tanto clientes de supermercado, quanto o próprio supermercado. Ele consiste em um aplicativo onde o usuário posiciona seu celular em um determinado produto e recebe na tela do próprio aparelho o valor do mesmo produto em outros supermercados, com isso saberá se valerá a pena adquirir o produto do supermercado em que está. Já o segundo módulo, é um site em que o usuário irá procurar o produto e receber o preço do mesmo em vários mercados, com a vantagem de não precisar ir ao mercado para fazer a pesquisa, já que de dentro de sua casa, saberá em qual mercado compensa comprar determinada mercadoria.



A importância do sistema consiste em duas partes: a otimização do tempo do cliente, pois o sistema terá uma pesquisa rápida trazendo o seus gastos, informando assim exatamente o local que deve ir comprar por um preço melhor. Já o lado do supermercado, tornará mais fácil e rápido o acesso do preço da concorrência, e com isso terá como avaliar o seu preço com os demais daquela determinada região, visando sempre à satisfação de seus clientes.

## **1.2. Objetivo**

O objetivo geral desse sistema é facilitar a vida de clientes/supermercados de um jeito inovador, trazendo o valor de um determinado produto pesquisado nos demais supermercados da região de uma maneira simples, rápida e econômica para ambas as partes. Facilitando assim, a pesquisa pelo melhor preço ao cliente e para a empresa uma atualização sobre o valor da concorrência.

## **1.3. Justificativa**

Atualmente, para verificar o valor de produtos, o cliente precisa juntar uma quantidade de panfletos de concorrentes ao supermercado que costuma ir, e dirigir-se ao local, ainda há a necessidade de comparar as ofertas com as explícitas nos panfletos, pois ainda existe a alteração de valores. Desta forma, com o aplicativo em mãos, ocorrerá a automatização de toda essa parte e incômodo de procurar panfletos, carregá-los e ter que se dirigir até os mercados para verificar os preços.

## **1.4. Método de Desenvolvimento**

Será usado o programa *Jude* e o *Astah Professional* para elaboração da arquitetura do software que compreende os casos de uso, diagrama de classe, diagrama de entidade relacionamento (DER), diagrama de atividade, diagrama de sequência e diagrama de Gantt.

Para elaboração do Software do celular, será usado a IDE Eclipse IDE for Java EE Developers, da fundação *The Eclipse Foundation*, com o plugin ADT que faz a comunicação com o SDK do *Android*.

Para fazer a comunicação com o banco de dados, terá um Webservice feito em Java que será responsável pela troca de dados entre o celular e o banco de dados, que em questão será usado o *PostgreSQL* versão 9.0.7 ou superior. E o site será feito em Java Web(JSF 2.0 +Prime Faces) ou asp.net (a definir).

A seguir será demonstrado os passos a ser seguidos para o Desenvolvimento do projeto.

- Levantamento do Material;
- Pesquisa de Trabalhos Correlatos;
- Desenvolvimento do Pré-Projeto;
- Mapa Mental;
- WBS;
- Sequenciamento de Atividades;
- Cronograma;
- Diagrama de Entidade Relacionamento(DER);
- Diagrama de Classes;
- Abstração de Casos de Uso;
- Diagrama de Sequência
- Especificação de Caso de Uso;
- Diagrama de Atividade;
- Criação do Banco de Dados;
- Desenvolvimento do Código e Interface;
- Testes;
- Implantação.

## 2. Tecnologias para Desenvolvimento

### 2.1. Android

Com a grande disseminação do Sistema Operacional iOS da *Apple*, observou-se, um novo espaço para um grande sistema operacional, de caráter livre, ao contrário do S.O da Apple que roda somente em hardwares próprios. Um grupo de desenvolvedores estavam com um projeto de implementar um S.O que fosse totalmente livre sua comercialização, tanto da parte do próprio S.O, como o lado dos desenvolvedores que teriam acesso fácil para produzir e disseminar aplicativos pra o tal.

Em Outubro de 2003 é fundada em Palo Alto, Califórnia, EUA; a *Android Inc.*, por Andry Rubin, Rich Miner, Nick Sears e Chris White. Com a repercussão que estava proporcionando, o Google resolve adquirir a empresa em Agosto de 2005. Após dois anos de trabalho e várias empresas vendo o real potencialmente que estaria por surgir, é criado em 5 de Novembro de 2007 o Consórcio *Open Handset Alliance* no qual será disposto abaixo mais especificado. Já em 12 de Novembro de 2007 após sete dias do consorcio formado, é liberado o tão aguardado SDK para os desenvolvedores, era a primeira versão de muitos que veriam e estão por vir. A *HTC* com o seu celular Dream (G1) é a primeira empresa a lançar um celular com *Android* em 22 de Outubro de 2008, da qual sua versão era a 1.0. A partir disto, as atualizações começaram a ser constantes. No ano de 2009, foi atualizado o *Kernel* do Linux, do qual falaremos mais abaixo, da versão 2.6.27 para 2.6.29 e a versão do *Android* foi 1.1, 1.5, a partir da versão 1.5 começaram a ser batizadas com nomes de doces americanos, veio ainda o 1.6 e a versão 2.0 encerrou o ciclo de atualizações da data coerente (CURA, 2011).

No ano de 2010 foram apenas três atualizações, uma de *Kernel* e uma do *Android*, como podemos observar, o *Android* já estava começando a se estabilizar, diferente do ano anterior em que houve bastantes atualizações; o *Kernel* passou para 2.6.32 em 20 de Maio, e dia 6 de Dezembro o *Android* chegou a versão 2.3 que é a mais utilizada até o momento (GARATTONI, 2010).

Em 2011 tivemos um grande marco para a comunidade *Android*, a versão 2.3 recebeu pequenas alterações passando para 2.3.3 e em 22 de Fevereiro com a alta dos Tabletes, é liberada a versão 3.0 destinado exclusivamente a eles, após dois meses é lançada no mercado a versão 3.1.

O principal lançamento do ano foi a versão do *Android* 4.0 aonde propunham uma integração e padronização da plataforma, tanto os Smartphones e Tabletes, puderam utilizar a mesma versão, sem preocupação com tamanhos de telas e hardwares; e no dia 19 de Outubro é liberado para as operadoras de telefonia e usuários o *Android* 4.0.

A Imagem abaixo ilustra a porcentagem de cada *Android* no mercado.

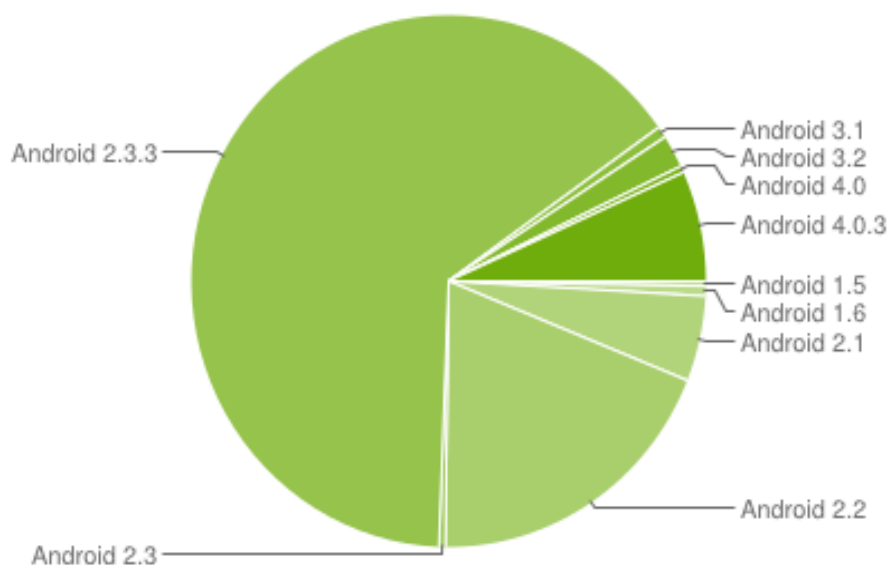


Figura 1: Porcentagem dos tipos de *Android* utilizados

**Fonte:** <http://developer.android.com/about/dashboards/index.html>

### 2.1.1. Plataforma Android

A figura a seguir demonstra as 5 camadas da arquitetura *Android*.

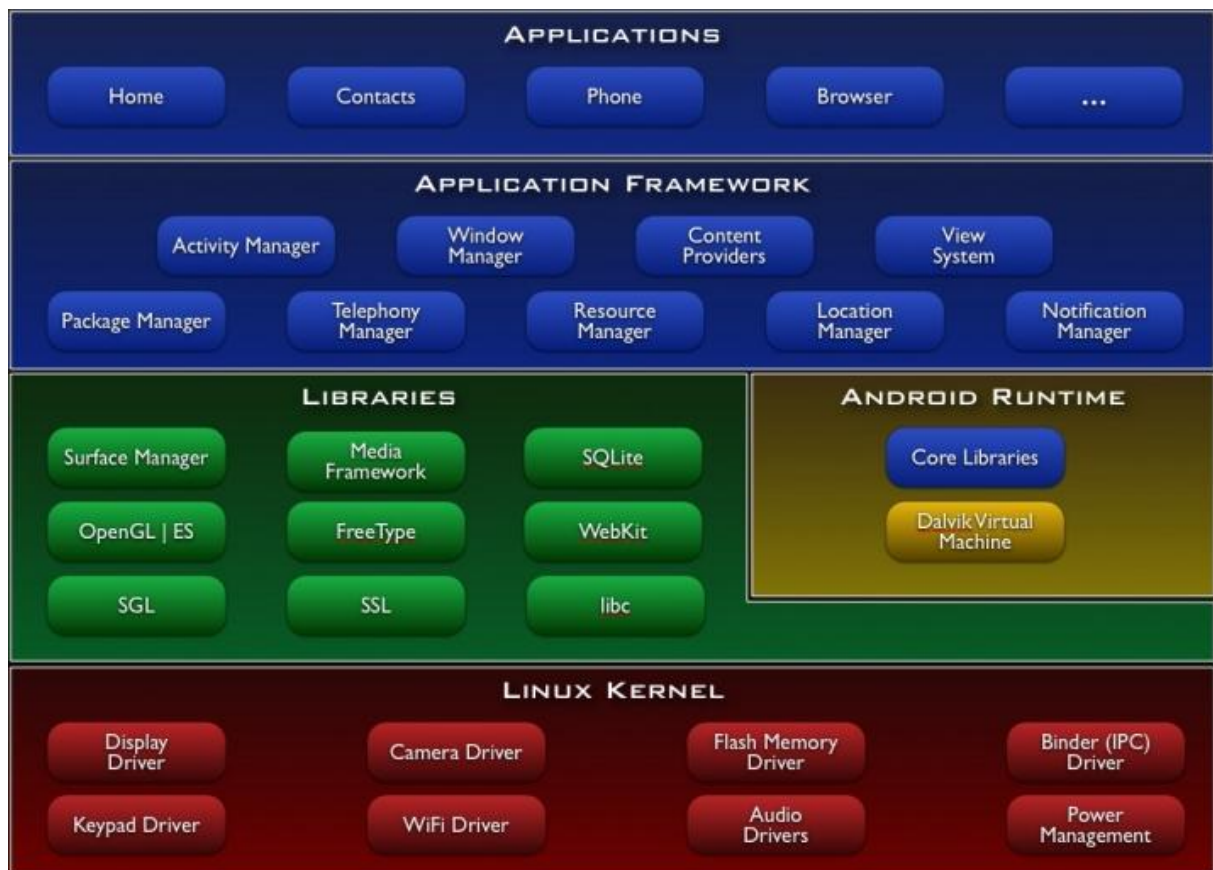


Figura 2: Arquitetura *Android*

Fonte: <http://www.sergioprado.org/wp-content/uploads/system-architecture.jpg>

- **Camada de Aplicação**

É a última camada da arquitetura, aonde situa-se os aplicativos tanto nativos como de terceiros, dentre eles se encontra-se os softwares de agenda, contatos, calculadora entre outros.

- **Framework de Aplicação**

É uma camada de alto nível da arquitetura, que facilita a produtividade e o desempenho das próprias aplicações nativas e de terceiros, a sua arquitetura foi modelada para que fosse possível reutilizar os recursos do S.O, qualquer aplicação consegue acessar eles, podemos citar o,

- Activity Manager: Lida com o ciclo de vida da Aplicação
- Windows Manager: Responsável pelas janelas de todas as aplicações.
- Content Providers : Provedores de Conteúdo que liberam o acesso a outros aplicativos para fazer um junção em um próximo aplicativo.

- **Bibliotecas**

Conjunto de Bibliotecas em C/C++ que auxiliam em vários componentes do framework.

**Surface Manager:** Trabalha com diferentes serviços de desenho, como as camadas gráficas 2D e 3D dos aplicativos; gerencia o acesso ao subsistema do display;

**OpenGL ES :** Sistema gráfico baseado no OpenGL ES 1.0;

**SqlLite:** Banco de Dados mobile.

- **Kernel Linux**

Camada responsável por controlar o hardware do celular/tablete, se encarrega de controlar o driver de Wireless, Áudio Drivers, Câmera Drivers, Bluetooth Driver, além da Segurança, Gerenciamento de Memória, Processos.

- **Android Runtime**

**De acordo com Cura, 2011:**

Toda aplicação *Android* roda em seu próprio processo, com sua própria instância da máquina virtual Dalvik. O Dalvik foi escrito de forma a executar várias VMs eficientemente. Ele executa arquivos .dex, que é otimizado para consumo mínimo de memória. A VM é baseada em registros e roda classes compiladas pela linguagem Java que foram transformadas em arquivos. dex, através da ferramenta “dx” incluída no SDK. Praticamente todas as bibliotecas oferecidas em Java, são compatíveis no *Android* (CURA, 2011).

**2.1.2. Android e seus nomes**

Uma tradição adotada pela Google foi batizar as versões *Android* com nome de doces, a primeira a receber foi a 1.5, o software foi batizado de *Cupcake*. A letra “C”, inicial do doce, é também a terceira do alfabeto. Desde então essa ordem vem sendo mantida e todas as outras versões do SO ganharam nomes de doces: *Donut*, *Eclair*, *Froyo*, *Gingerbread* e *Honeycomb*, *Ice Cream Sandwich*, que podemos notar que cada palavra à primeira ordem segue a ordem do alfabeto.

- Versão 1.5 – *Cupcake* (Bolo de Caneca)
- Versão 1.6 – *Donut* (Rosquinha)
- Versão 2.0 – *Éclair* (Bomba de Chocolate)
- Versão 2.2 – *Froyo* (Iogurte Congelado)
- Versão 2.3 – *Gingerbread* (Pão de Gengibre)
- Versão 3.0 – *Honeycomb* (Favo de Mel)



- Versão 4.0 – *Ice Cream Sandwich* (Sanduiche de Sorvete)
- Versão 4.1 - *Jelly Bean*.

E fontes já dizem que a próxima versão vira com o nome de *Key Lime Pie*.

### 2.1.3. Open Handset Alliance (OHA)

Como o *Android* é livre, licenciado sobre a licença *Apache Software Foundation* (ASF) 2.0 (Que é uma licença permissiva de software livre sem *copyleft*) (Torres, 2012), poderia ocorrer inconsistência por parte das empresas, e o que tornaria prejudicial quando fosse haver atualizações para várias versões *Android* de várias empresas, pensando nisso criou-se a *Open Handset Alliance* é uma aliança de várias empresas que se formaram para padronizar a plataforma *Android*.

A seguir segue uma imagem, com as empresas separadas pelas suas respectivas categorias que compõe a Aliança - *Open Handset Alliance*.

Operator	Handset Makers	Software Companies	Commercialization Companies	Semiconductor Companies
       	         	           	     	            

Figura 3: As empresas que compõe a Aliança - Open Handset Alliance

Fonte: <http://sbroh.files.wordpress.com/2009/11/eab7b8eba6bc-1.png>



## 2.2. Java

Java é uma linguagem 100% orientada a objetos, mas sua base se deu através do C++, de acordo com DEITEL (2003), seu desenvolvimento teve início, em 1991, na *Sun Microsystem* em um projeto chamado codinome Green (DEITEL, 2003).

O intuito principal era criar uma tecnologia de desenvolvimento de software para aparelhos domésticos (como podemos ver que hoje em dia, Java, esta embarcado em vários aparelhos como geladeiras, fogões, fornos elétricos) e televisão interativa. James Gosling, o criador, a chamou de Oak em homenagem a uma árvore que ele observava sempre, mas após um tempo viram que já existia uma linguagem com esse nome, e tiveram que troca lá. Como a equipe de desenvolvimento tomavam muitas xicaras de café e precisavam de um nome, e batizaram a de Java lançado posteriormente em maio de 1995 (FREITAS, 2007).

De acordo com Pacievitch em 2011:

Em 2006 muitas partes do Java estavam sendo passadas para a licença de Software livre, e a maioria já estava disponível para o publico gratuitamente, tudo Sob licença GNU. O Java foi uma revolução na interatividade, sua utilização aumenta a cada dia. Java é uma linguagem relativamente simples e dinâmica, permite criar programas e aplicações para a Web sem depender de outra linguagem (PACIEVITCH, 2011).

### 2.2.1. Java Serve Faces (JSF)

Hoje em dia podemos escolher entre muitos frameworks para o desenvolvimento de interfaces de usuários para uma aplicação web. JavaServer Faces (JSF) é um framework baseado em componentes que nos permite a criação de UIs. Por exemplo, se quiséssemos exibir uma tabela com linhas e colunas, não seria necessário gerar tags de HTML para linhas e células como em um loop, basta adicionar um componente de tabela na página. Podemos dizer que JSF é baseado nas seguintes partes (GEARY, 2010):

- uma série de componentes pré-fabricados para criação de UIs (*User Interfaces*).
- Um modelo de programação baseado em eventos
- um modelo de componentes que permite qualquer usuário desenvolver seus próprios elementos adicionais.

JSF contém todo o código necessário para a manipulação de componente e organização de eventos. JSF não é o único framework web baseado em componentes, mas é a camada de visão no padrão Java EE. JSF está incluído em cada servidor de aplicativos Java EE, e pode ser facilmente adicionado a um container web como o *Tomcat*.

### 2.3. PostgreSQL

O sistema gerenciador de banco de dados (SGDB) objeto-relacional, denominado de PostgreSQL, possui em sua robusta estrutura código aberto. Detém mais de 15 anos de desenvolvimento atuante, e uma base estruturada de atributos como confiabilidade, integridade de dados e conformidade a padrões.

Também possui total compatibilidade com todos os grandes sistemas operacionais, tais quais: GNU/Linux, Unix (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), e MS Windows.

O PostgreSQL é conhecido pelo fácil suporte e integração a chaves estrangeiras, junções, visões, gatilhos e procedimentos armazenados. Nele, faz-se presente

grande quantidade de dados do ISO SQL:1999, como INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATA, INTERVAL, e TIMESTAMP. Além de admitir alocação de objetos binários, abrangendo figuras, sons ou vídeos.

## **2.4. Metodologias e Arquiteturas**

### **2.4.1. Casos de Uso**

Os casos de uso são utilizados para que a comunicação entre o cliente e o analista se torne mais fácil, proporcionando uma visão do cenário, o qual mostra as funcionalidades do sistema a partir da visão do usuário (SAMPAIO, 2007).

### **2.4.2. Diagrama de Classes**

O diagrama de classes é definido para exibir como a estrutura do sistema, levando em consideração o conceito de classes e suas relações. Classe é a representação de um objeto, tendo os seus atributos e comportamentos tendo como ligação as relações que são as ligações entre as classes. ([http://www.fca.pt/docs-online/722-636-8\\_des\\_sis\\_inf.pdf](http://www.fca.pt/docs-online/722-636-8_des_sis_inf.pdf))

### **2.4.3. Diagrama de Entidade Relacionamento**

Definidos para aperfeiçoar a percepção do mundo real para o mundo virtual, tendo como objetos as entidades e para interligar estas, os relacionamentos.

### **2.4.4. Diagrama de Atividades**

O Diagrama de Atividade foi pensado para mostrar como uma atividade dependa da outra na execução de um processo, no qual as mesmas estão dispostas.

### **2.4.5. Sequenciamento de atividades**

O sequenciamento de atividades, objetiva, identificar e documentar os relacionamentos lógicos entre as atividades presentes no cronograma. Para que se possa obter um cronograma com característica real e atingível, é preciso que as atividades estejam sequenciadas corretamente.

De acordo com Junior (2002), o sequenciamento pode ser realizado com o suporte de um computador (programas computacionais), ou com métodos manuais. Onde, as técnicas manuais podem ser mais eficientes quando aplicadas a projetos menores, ou projetos maiores que não possuem alto nível de detalhamento. Contudo, pode-se utilizar os métodos manuais e automatizados em conjunto.

### 3. Documentação

A documentação é uma das partes mais importantes em um desenvolvimento de um Sistema Web ou Aplicações Mobile. Os tópicos a seguir, irão demonstrar todos os componentes que a compõe:

#### 3.1. Mapa Mental

O Mapa Mental é um diagrama que esboça as principais ideias referentes ao projeto, ele não precisa ter todo o mecanismo do projeto demonstrado, mas sim, do ponto de vista que olhando, possa passar o qual a ideia e o intuito do sistema.

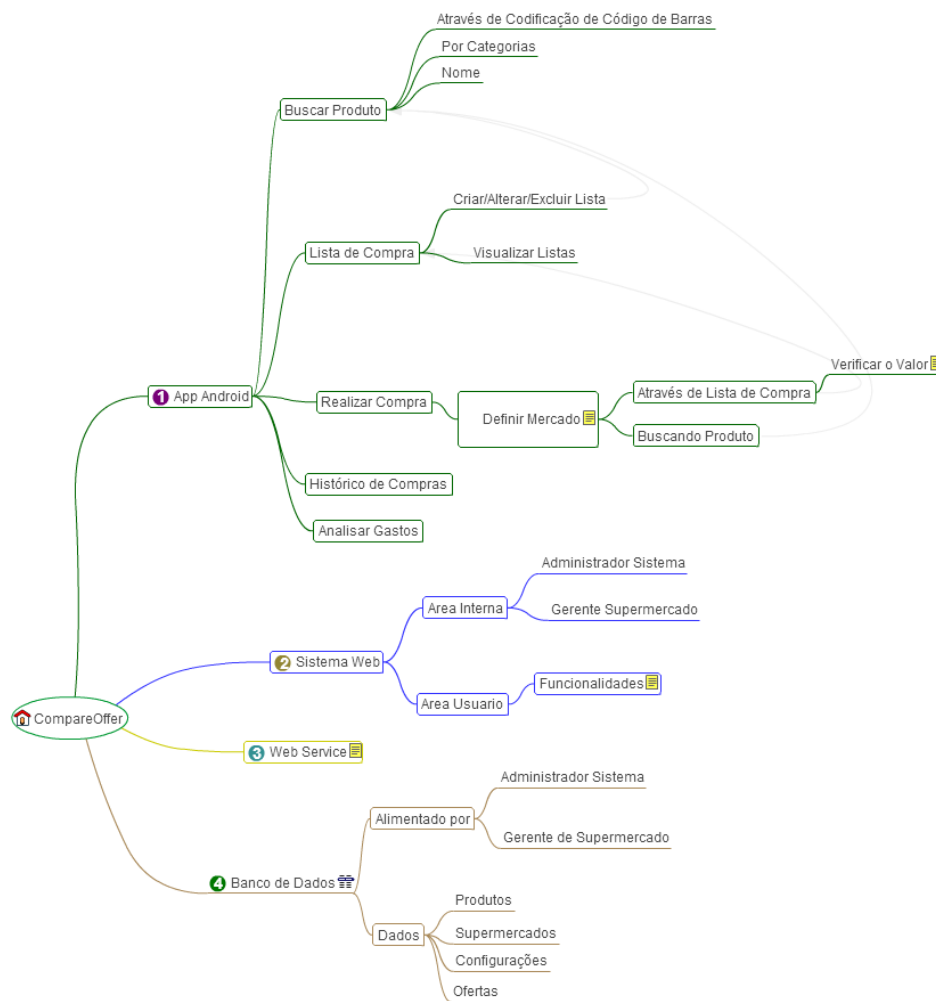


Figura 4 - Mapa Mental

### 3.2. WBS

A Figura seguir demonstra a estrutura do planejamento do projeto, começando desde o Levantamento de Requisitos com o cliente, até o momento da implantação no próprio, dividindo as etapas em nós (categorias) e ramificações (SubCategorias).

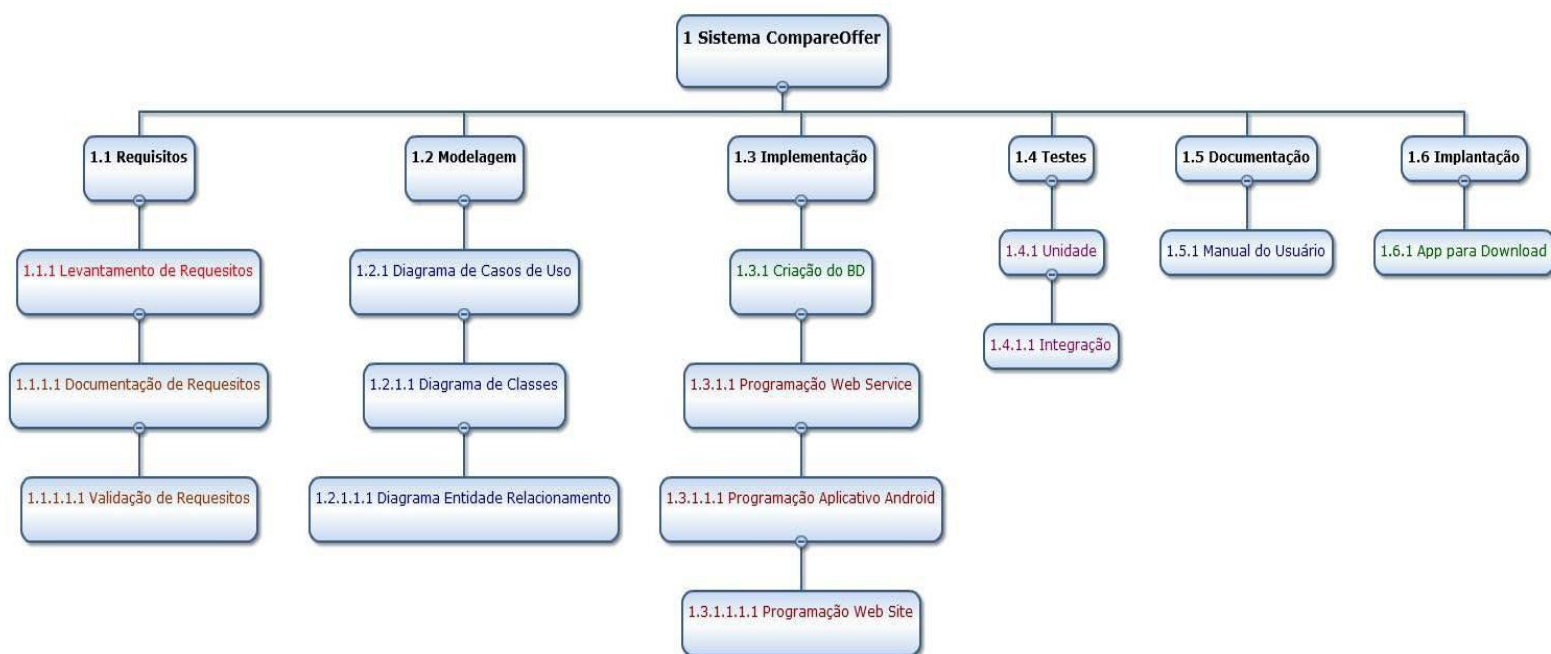


Figura 22: WBS<sup>1</sup>

<sup>1</sup> Work breakdown structure. <http://host.utp.br/?p=321>

### 3.3. Sequenciamento de Atividades

A Figura 4 demonstra o sequenciamento de atividades, que traz em si, um retrospecto das atividades a seguir com o seu tempo previsto, desde uma sequencia de atividades, como atividades em paralela.

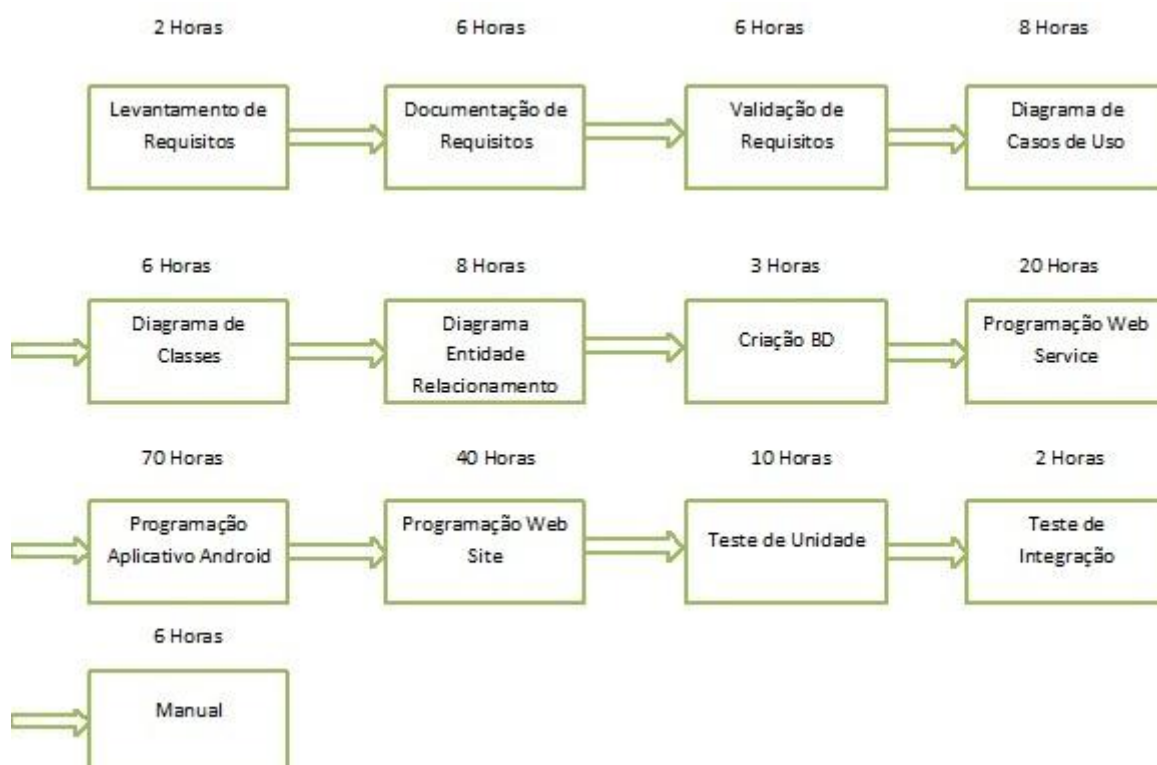


Figura 5: Sequenciamento de Atividade

### 3.4. Cronograma

Com o cronograma, temos uma visão parcialmente das atividades que serão realizadas, em seu determinado tempo, e com isso, há a possibilidade de acompanhar se tal atividade esta dentro do seu prazo, ou se estourou o tempo necessário para realização da própria.

CRONOGRAMA	FEVEREIRO		MARÇO		ABRIL		MAIO		JUNHO		JULHO		AGOSTO		SETEMBRO		OUTUBRO		NOVEMBRO	
	01 a 15	16 a 29	01 a 15	16 a 31	01 a 15	16 a 30	01 a 15	16 a 31	01 a 15	16 a 30	01 a 15	16 a 31	01 a 15	16 a 31	01 a 15	16 a 30	01 a 15	16 a 31	01 a 15	16 a 30
Levantamento de Material																				
Pesquisa de Trabalho Correlatos																				
Desenvolvimento Pré Projeto																				
Levantamento de Requisitos																				
Mapa Mental																				
Abstração de Casos de Uso																				
Especificações de Caso de Uso																				
Diagrama de Classe																				
Diagrama Ent Relacionamento																				
Diagrama de Atividade																				
Diagrama de Sequencia																				
Entrega TCC para Qualificação																				
Criação do Banco de Dados																				
Programação																				
Testes																				
Envio/Volta TCC Orientador																				
Apresentação Final																				

Figura 6: Cronograma



### 3.5. Diagrama Entidade Relacionamento (DER)

A figura 6 demonstra o digrama de Entidade Relacionamento que demonstra as tabelas do BD com as suas respectivas ligações.

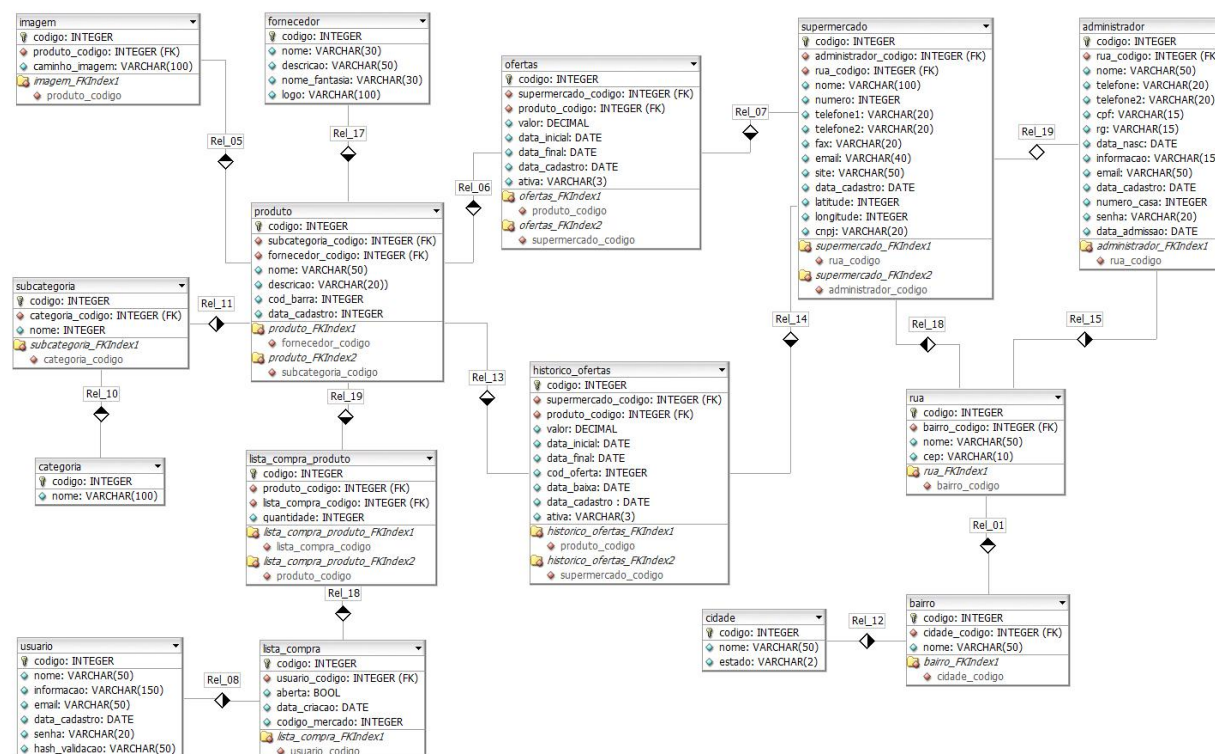


Figura 7: Diagrama Entidade Relacionamento (DER)

### 3.6. Diagrama de Classes

Designado para demonstrar a representação da estrutura do projeto, dividindo cada tipo de classe no seu pacote, e ter uma boa relação entre os tipos de classe.

#### 3.6.1. Pacote Geral

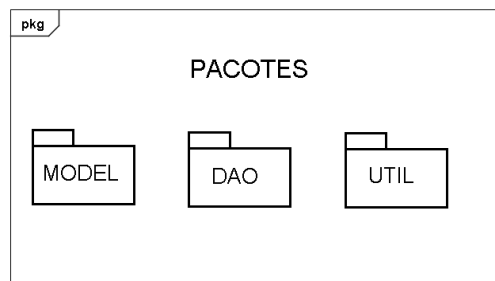


Figura 8: Pacote Geral

#### 3.6.2. Pacote Útil

O pacote Útil é responsável pelas classes auxiliares, mais não menos importante no software, dentre elas temos a de Conexão, responsável por toda a conexão ao banco, uma próxima classe chamada validações que conte métodos auxiliares, como validar data, trazer data do dia entre outras.

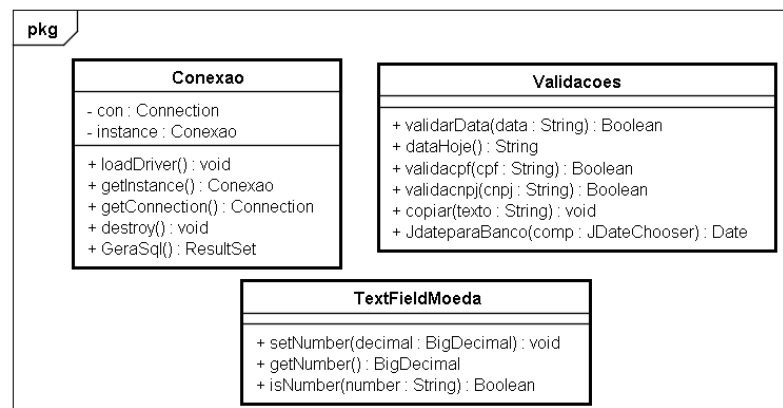


Figura 9: Pacote Útil

### 3.6.3. Pacote Model

Pacote Model (Modelo) responsável pelas classes modelo do sistema, eles serão mais usados geralmente no DAO.

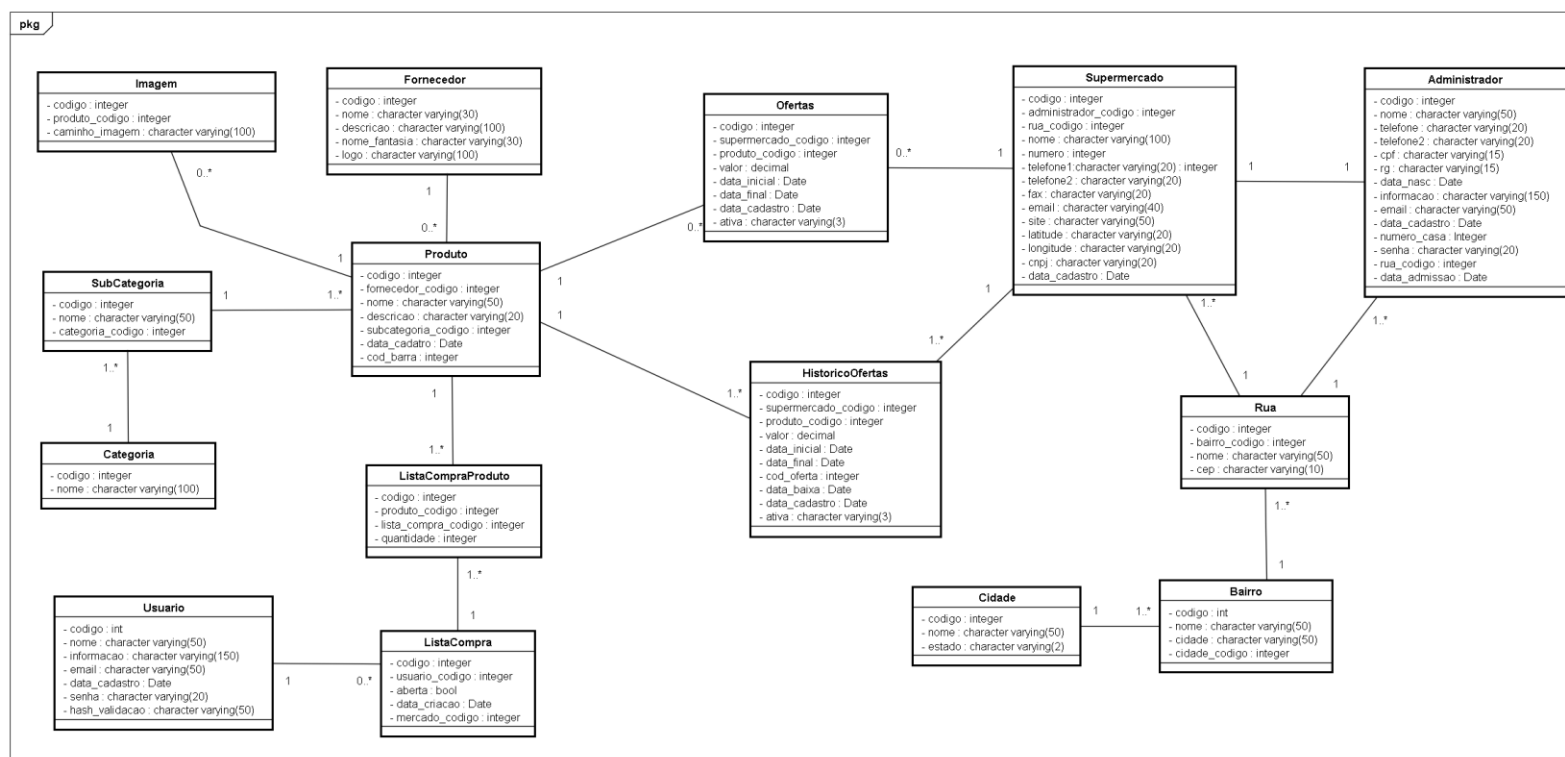


Figura 10: Diagrama de Classes

Avenida Getúlio Vargas, 1200 - Vila Nova Santana - Assis/SP - Cep 19807-634  
Fone/Fax (18) 3302 1055 - Homepage : [www.fema.edu.br](http://www.fema.edu.br)

### 3.6.4. Pacote DAO

Pacote DAO (Modelo) responsável pelas classes dos métodos do sistema, boa parte dos métodos é usado as classes MODEL.

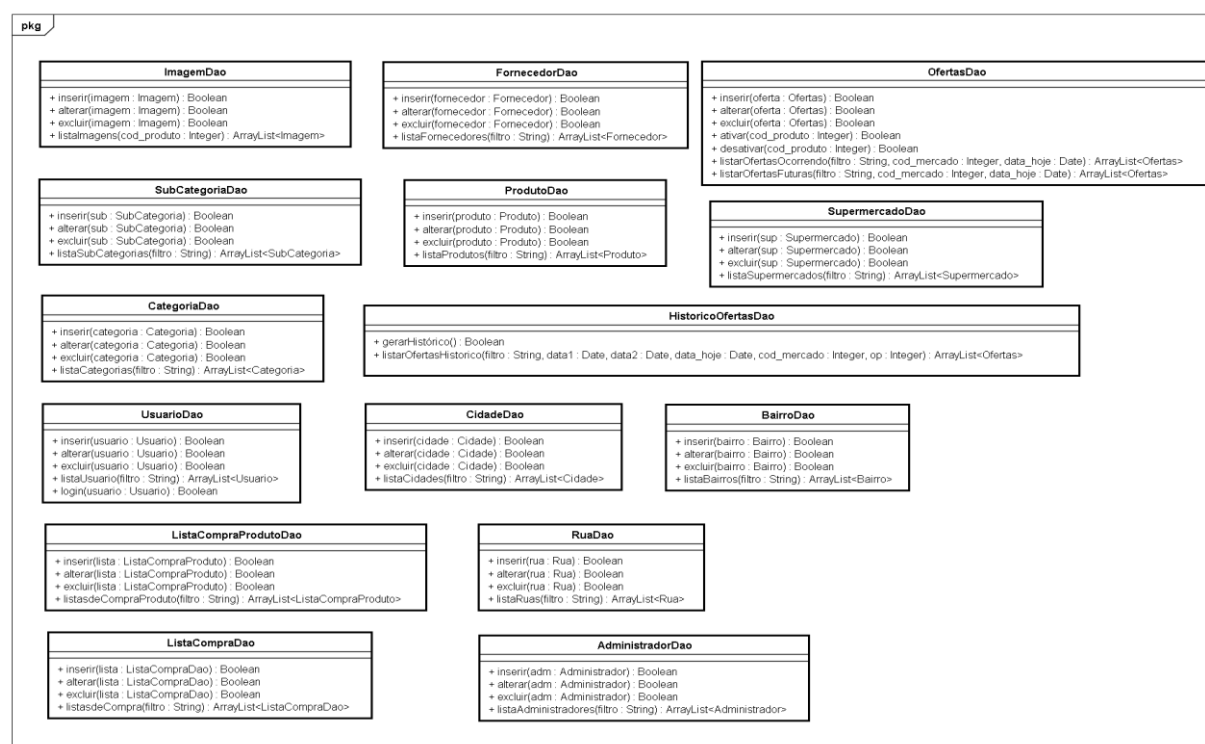


Figura 11: Pacote DAO

Avenida Getúlio Vargas, 1200 - Vila Nova Santana - Assis/SP - Cep 19807-634  
Fone/Fax (18) 3302 1055 - Homepage : [www.fema.edu.br](http://www.fema.edu.br)

### 3.7. Casos de Uso

No Caso de Uso geral do Projeto que envolve o sistema *Android* e o sistema complementar, partindo dele o Administrador do sistema terá acesso a vários módulos complementares que ajudarão na desenvolvimento do projeto, como os módulos “manter” de cidade, bairro, rua, supermercado, administrador do supermercado, produto, fornecedor aonde poderá cadastrar, alterar, excluir e visualizar todos os dados referentes. Terá a sua disponibilização relatórios como de supermercados, fornecedores e produtos.

O trabalho referente ao administrador do supermercado será o acesso ao módulo Manter Oferta, o qual é responsável por definir quais serão as ofertas válidas para o supermercado. O seu funcionamento é simples, irá aparecer na tela todos os produtos existentes cadastrados no sistema (caso não possua algum, ele poderá entrar em contato e pedir a adição de algum produto) irá escolher o produto e definir quanto custa o respectivo em seu supermercado, terminando a lista com todas as ofertas, será definido a data de início e a data final que ocorrerão.

As funcionalidades do usuário *Android*, será criar uma lista de compras para facilitar quando estiver em um supermercado para realizar uma compra. Outra funcionalidade é realizar uma compra virtual, com base em vários supermercados, fazendo uma pesquisa prévia, assim, obterá o menor custo possível. Para complementar esta última funcionalidade, ele poderá imprimir essa lista de compras para facilitar sua locomoção aos supermercados.

E o módulo mais importante do aplicativo, é o de busca de produtos, onde ele visualizará no celular, estando ou não em um supermercado, apenas apontando a câmera do celular com o programa aberto para o produto com código de barras. O mesmo mostrará o produto e o valor dele em outros supermercados (que cadastraram suas ofertas), tendo em suas mãos se compensa ou não a comprar deste produto, no supermercado em questão.

Na figura 12 temos o diagrama que demonstrará isso.

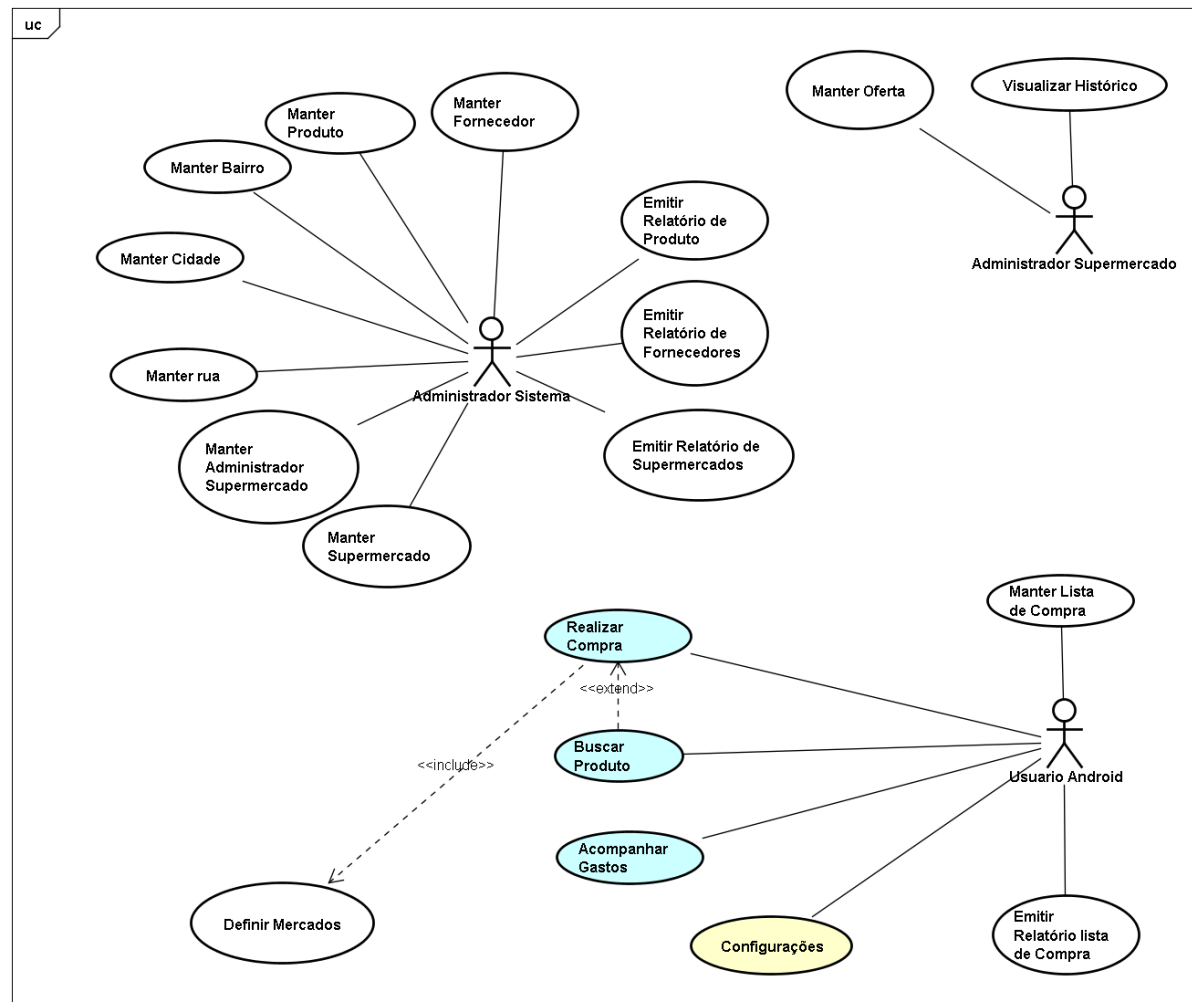


Figura 12: Diagrama de Casos de Uso

### 3.7.1. Especificações do Ator Administrador Supermercado

As especificações dos casos de uso referente ao ator Administrador Supermercado estão dispostas abaixo, sendo duas, Manter Oferta e Visualizar Histórico:

### 3.7.1.1. Diagrama de Sequência – Manter Oferta

Nesse caso de uso o administrador do supermercado irá gerenciar todas as ofertas que serão destinadas ao público, através do aplicativo *Android*. Poderá ser adicionada, alterada ou excluída uma oferta a qualquer momento.

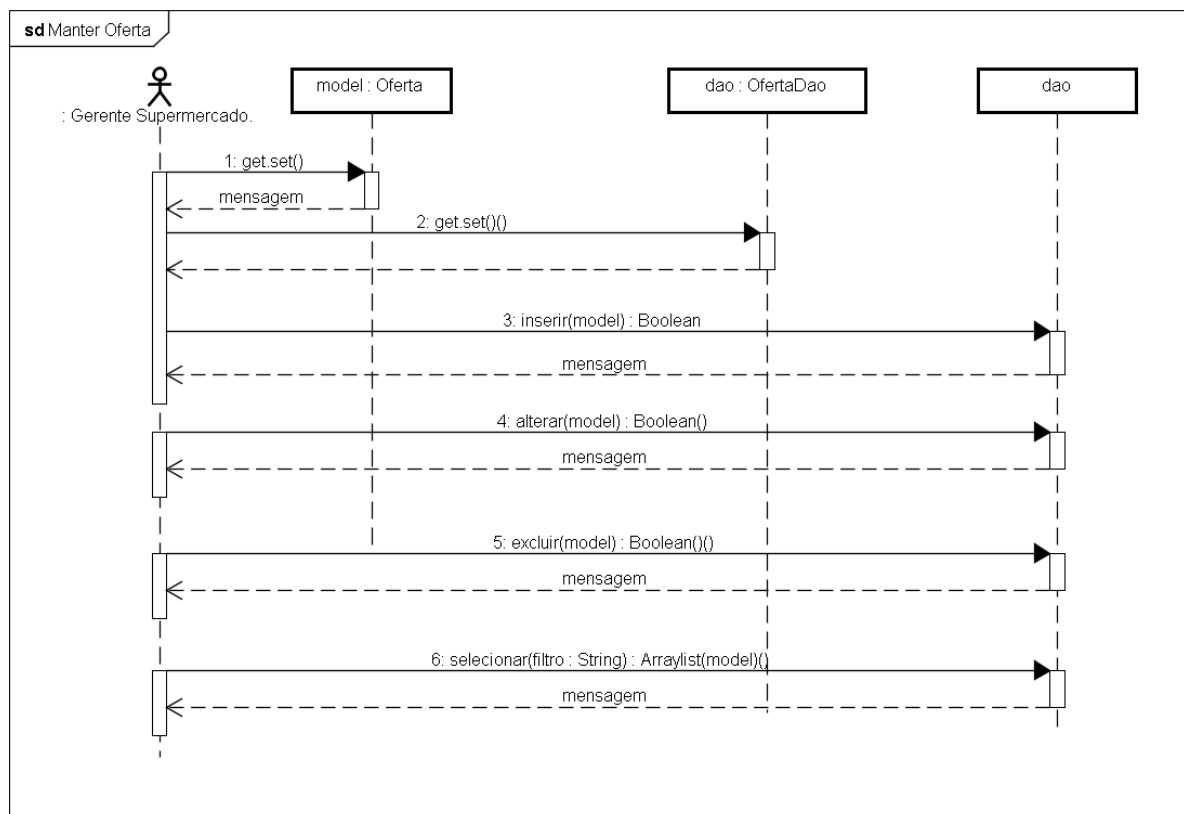


Figura 13: Diagrama de Sequencia - Manter Oferta

#### 3.7.1.1.1. Manter Oferta - Use Case 1

Na figura 14, será demonstrado o caso de uso “*Manter Oferta*” responsável por cadastrar todas as ofertas, módulo que será consumido pelos administradores dos supermercados, geralmente o gerente ou alguém do mesmo patente ou superior.

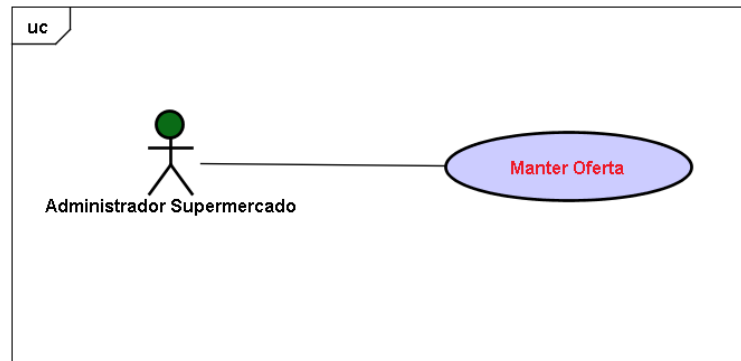


Figura 14: Manter Oferta - Use Case 1

Tabela 1: Especificação de Use Case 1 - Manter Oferta

<b>Nome da Use Case 1</b>	Manter Oferta
<b>Ator (es)</b>	Administrador Supermercado
<b>Pré-Condição</b>	O administrador supermercado deverá estar autenticado no sistema.
<b>Cenário Principal</b>	1- O sistema solicita os dados necessários para cadastrar a oferta. 2- O administrador supermercado informa os dados de acordo com os campos a serem preenchidos. 3- O sistema solicita os dados para o cadastro da função. 4- O administrador supermercado informa os dados necessários. 5- O administrador supermercado seleciona a opção Cadastrar 6- O sistema emite a mensagem "Oferta cadastrada com sucesso". 7- O sistema cadastra a oferta.
<b>Cenário Alternativo</b>	Se o administrador supermercado não informar os dados para o cadastro da função, o sistema informa



	<p>que a oferta não está cadastrada.</p> <p>O administrador supermercado poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.1.2. Diagrama de Sequência - Visualizar Histórico

Nesse caso de uso, o administrador do supermercado irá acessar todo histórico de ofertas que foram geradas, que já venceram seu prazo, podendo ver desde todas, como escolhendo por supermercado e filtrando por datas de pesquisa.

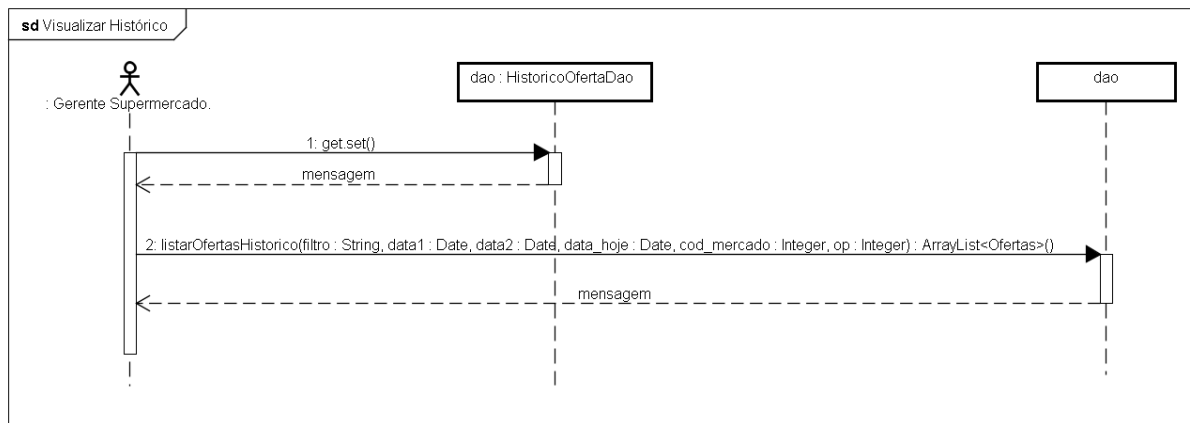


Figura 15: Diagrama de Sequência - Visualizar Histórico

### 3.7.1.2.1. Visualizar Histórico - *Use Case 2*

Caso de uso responsável por visualizar o histórico de todas as ofertas correspondente ao supermercado que o administrador estiver cadastrado. Com isso poderá fazer uma média dos valores aplicados aos produtos.

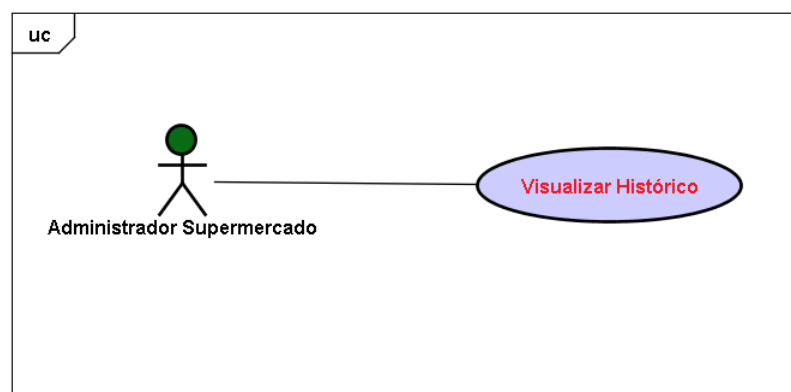


Figura 16: Visualizar Histórico - *Use Case 2*

Tabela 2: Especificação de *Use Case 2* - Visualizar Histórico

<b>Nome da <i>Use Case 2</i></b>	Visualizar Histórico
<b>Ator (es)</b>	Administrador Supermercado
<b>Pré-Condição</b>	O administrador supermercado deverá estar autenticado no sistema.
	1- O sistema disponibiliza os dados necessários para visualizar o histórico. 2- O administrador supermercado seleciona o botão Visualizar Histórico. 3- O administrador supermercado seleciona o botão Imprimir.

<b>Cenário Principal</b>	4- O sistema imprime o histórico com sucesso.
<b>Cenário Alternativo</b>	O administrador supermercado poderá visualizar o histórico e não imprimir.
<b>Casos de Teste</b>	4.1- O administrador supermercado cancela a operação.

### **3.7.2. Especificações do Ator Administrador Sistema**

Os casos de Uso do Ator Administrador do Sistema, representados posteriormente, são Manter Administrador Supermercado, Manter Rua, Manter Bairro, Manter Produto, Manter Supermercado Conveniado, Manter Fornecedor, Emitir Relatório de Supermercado, Emitir Relatório de Fornecedor e Emitir Relatório de Produto:

#### **3.7.2.1. Diagrama de Sequência - Manter Administrador Supermercado**

O Administrador do sistema irá gerenciar todos os outros administradores dos diferentes supermercados; podendo inserir, alterar e excluir um administrador, definindo em qual estabelecimento essa pessoa presta seus serviços.

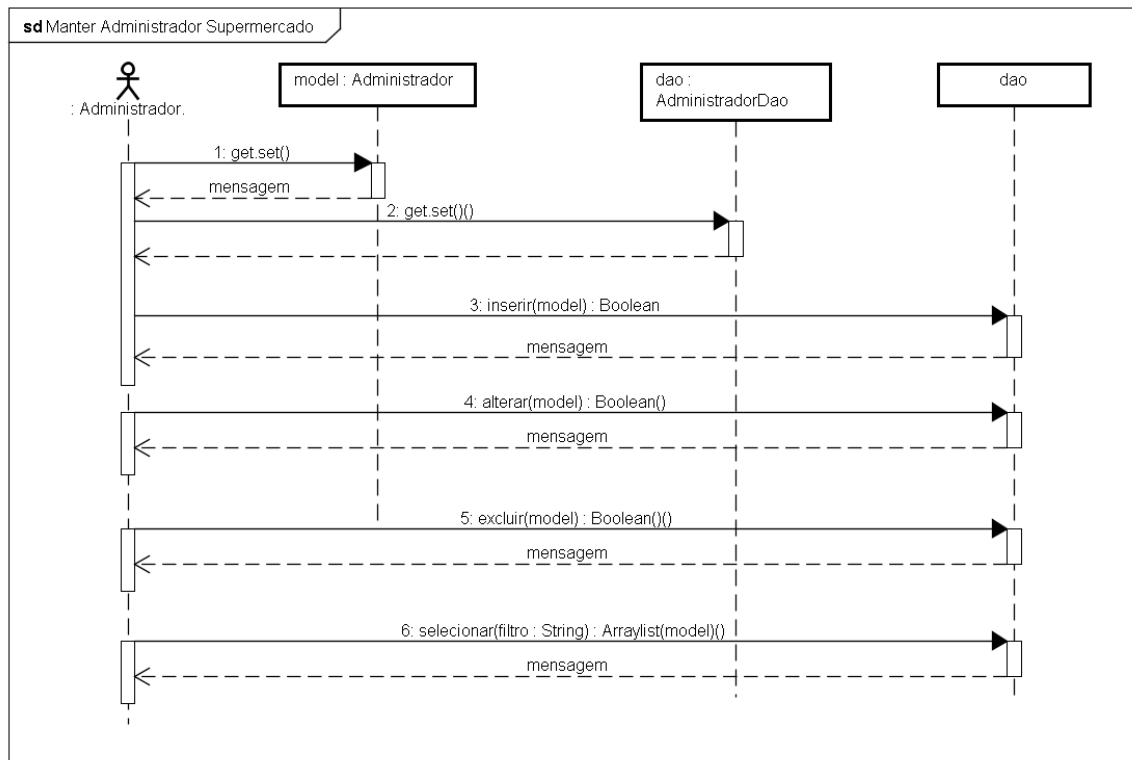


Figura 17: Diagrama de Sequência - Manter Administrador Supermercado

### 3.7.2.1.1. Manter Administrador Supermercado - Use Case 3

Este caso de uso é responsável pelo cadastro do administrador; pedindo vários dados como nome, RG, endereço, telefone entre outros dados cadastrais. E será usado quando for cadastrar um supermercado, sendo definido por eles um administrador.

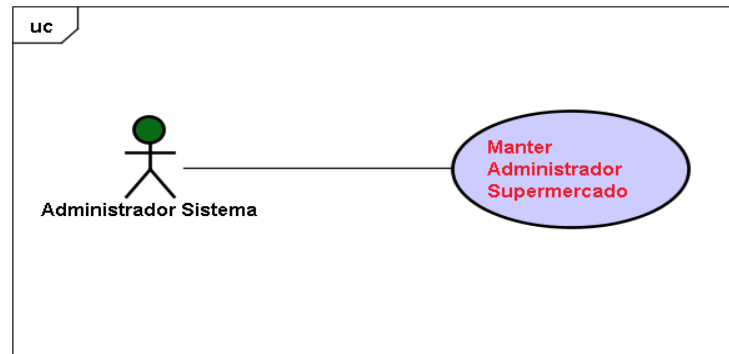


Figura 18: Manter Administrador Supermercado - Use Case 3

Tabela 3: Especificação de Use Case 3 - Manter Administrador Supermercado

<b>Nome da Use Case 3</b>	Manter Administrador Supermercado
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	<p>1- O sistema solicita os dados necessários para cadastrar o administrador supermercado.</p> <p>2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos.</p> <p>3- O sistema solicita os dados para o cadastro da função.</p> <p>4- O administrador sistema informa os dados necessários.</p> <p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem "Administrador supermercado cadastrado com sucesso".</p> <p>7- O sistema cadastra o administrador supermercado.</p>
<b>Cenário Alternativo</b>	Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que o

	<p>administrador supermercado não está cadastrado.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.2. Diagrama de Sequência – Manter Rua

Neste caso de Uso o Administrador do Sistema irá acessar o módulo de Ruas, para poder acrescentar, alterar, excluir e visualizar todas as ruas cadastradas.

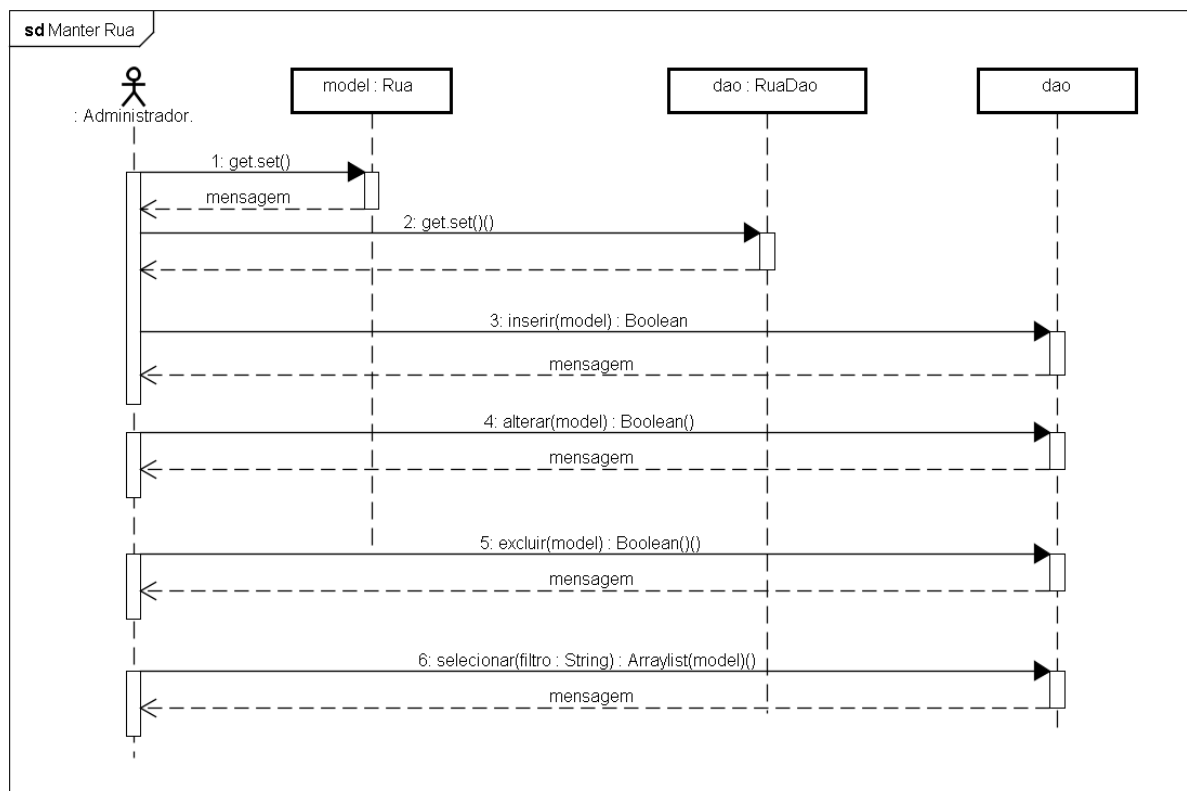


Figura 19: Diagrama de Sequência - Manter Rua

### 3.7.2.2.1. Manter Rua - Use Case 4

Na figura 20, vemos o caso de uso Manter Rua, aonde serão cadastradas todas as ruas existentes na cidade, para concluir o cadastro será necessário escolher um bairro.

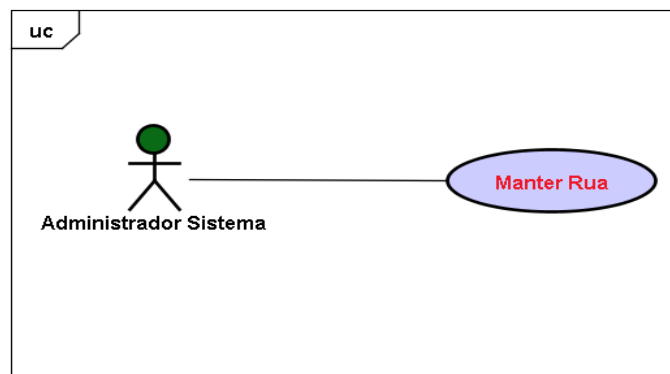


Figura 20: Manter Rua – Use Case 4

Tabela 4: Especificação de Use Case 4 - Manter Rua

<b>Nome da Use Case 4</b>	Manter Rua
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	1- O sistema solicita os dados necessários para cadastrar a rua. 2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos. 3- O sistema solicita os dados para o cadastro da função. 4- O administrador sistema informa os dados necessários.

	<p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem “Rua cadastrada com sucesso”.</p> <p>7- O sistema cadastra a rua.</p>
<b>Cenário Alternativo</b>	<p>Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que a rua não está cadastrada.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.3. Diagrama de Sequência – Manter Bairro

Neste caso de Uso, o Administrador do Sistema irá acessar o módulo de Bairro, para acrescentar, alterar, excluir e visualizar todos os bairros cadastrados.



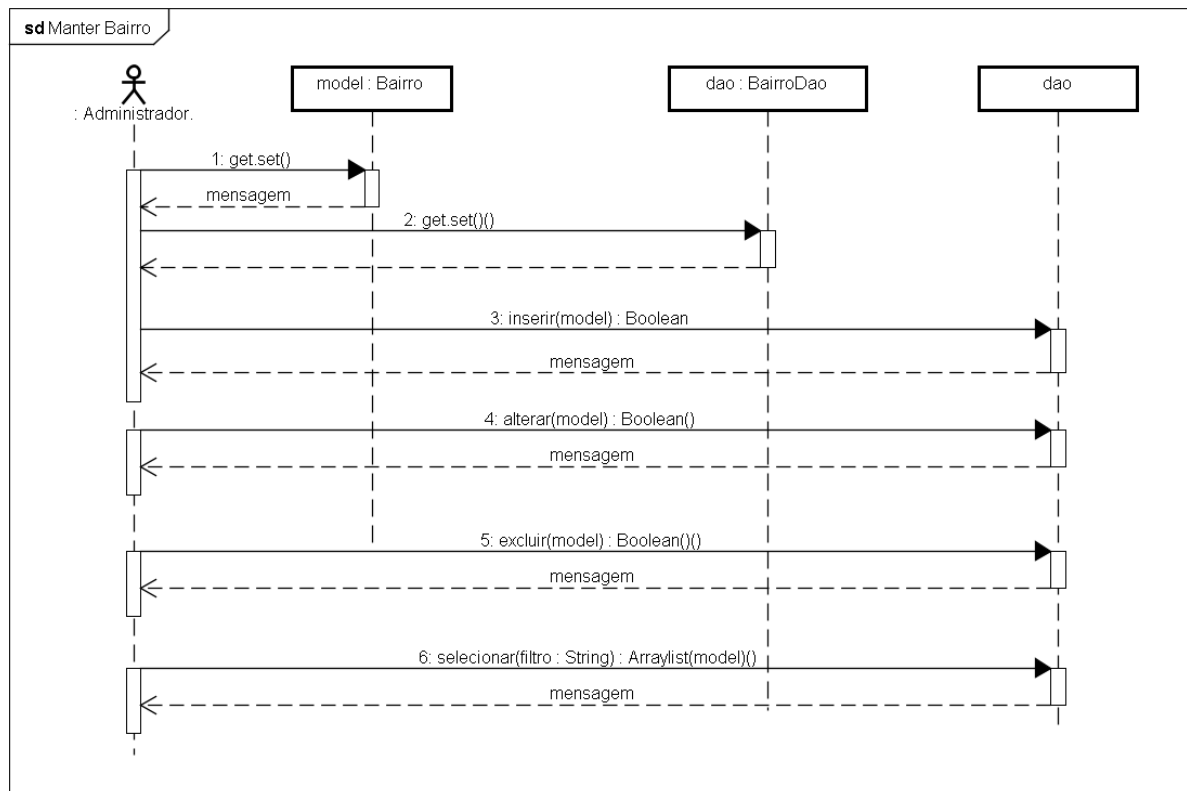


Figura 21: Diagrama de Sequência - Manter Bairro

### 3.7.2.3.1. Manter Bairro - Use Case 5

A figura 22, ilustra o caso de uso Manter Bairro, onde será cadastrado todos os bairros necessários para a inserção das ruas. Para concluir o cadastro, será necessário escolhido uma cidade.

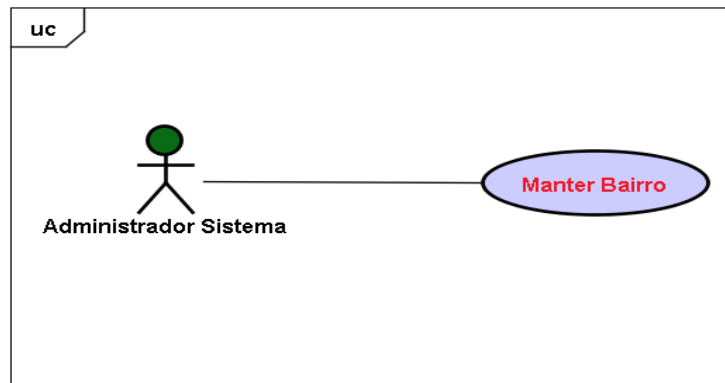


Figura 22: Manter Bairro - Use Case 5

Tabela 5: Especificação de Use Case 5 - Manter Bairro

<b>Nome da Use Case 5</b>	Manter Bairro
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	<p>1- O sistema solicita os dados necessários para cadastrar o bairro.</p> <p>2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos.</p> <p>3- O sistema solicita os dados para o cadastro da função.</p> <p>4- O administrador sistema informa os dados necessários.</p> <p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem "Bairro cadastrado com sucesso".</p> <p>7- O sistema cadastra o bairro.</p>

<b>Cenário Alternativo</b>	<p>Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que o bairro não está cadastrado.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.4. Diagrama de Sequência – Manter Cidade

Neste caso de Uso, o Administrador do Sistema acessará o módulo de Cidade, para poder acrescentar, alterar, excluir e visualizar todos os bairros cadastrados.

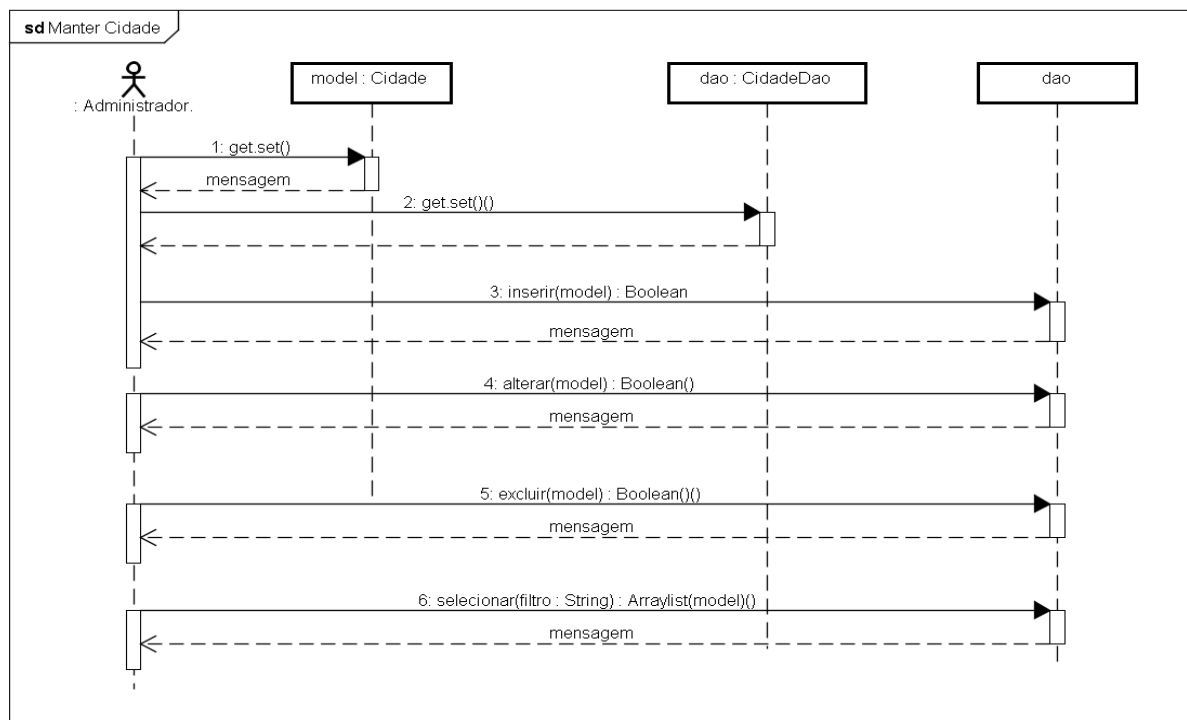


Figura 23: Diagrama de Sequência - Manter Cidade

### 3.7.2.4.1. Manter Cidade - Use Case 6

Manter Cidade, caso de uso responsável por cadastrar as cidades onde a supermercados conveniados ao sistema, e é usado também para o cadastro dos bairros.

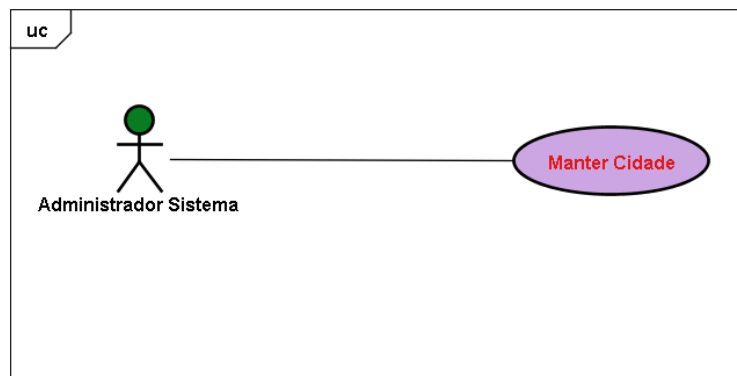


Figura 24: Manter Cidade - Use Case 6

Tabela 6: Especificação de Use Case 6 - Manter Cidade

<b>Nome da Use Case 6</b>	Manter Cidade
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	1- O sistema solicita os dados necessários para cadastrar a cidade. 2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos. 3- O sistema solicita os dados para o cadastro da função. 4- O administrador sistema informa os dados necessários.

	<p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem "Cidade cadastrada com sucesso".</p> <p>7- O sistema cadastra a cidade.</p>
<b>Cenário Alternativo</b>	<p>Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que o bairro não está cadastrado.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.5. Diagrama de Sequência - Manter Produto

Neste caso de Uso, o Administrador do Sistema acessará o módulo de Produtos, para acrescentar, alterar, excluir e visualizar todos os produtos cadastros. Essa base de dados servirá para que os administradores dos supermercados disponibilizem suas ofertas.

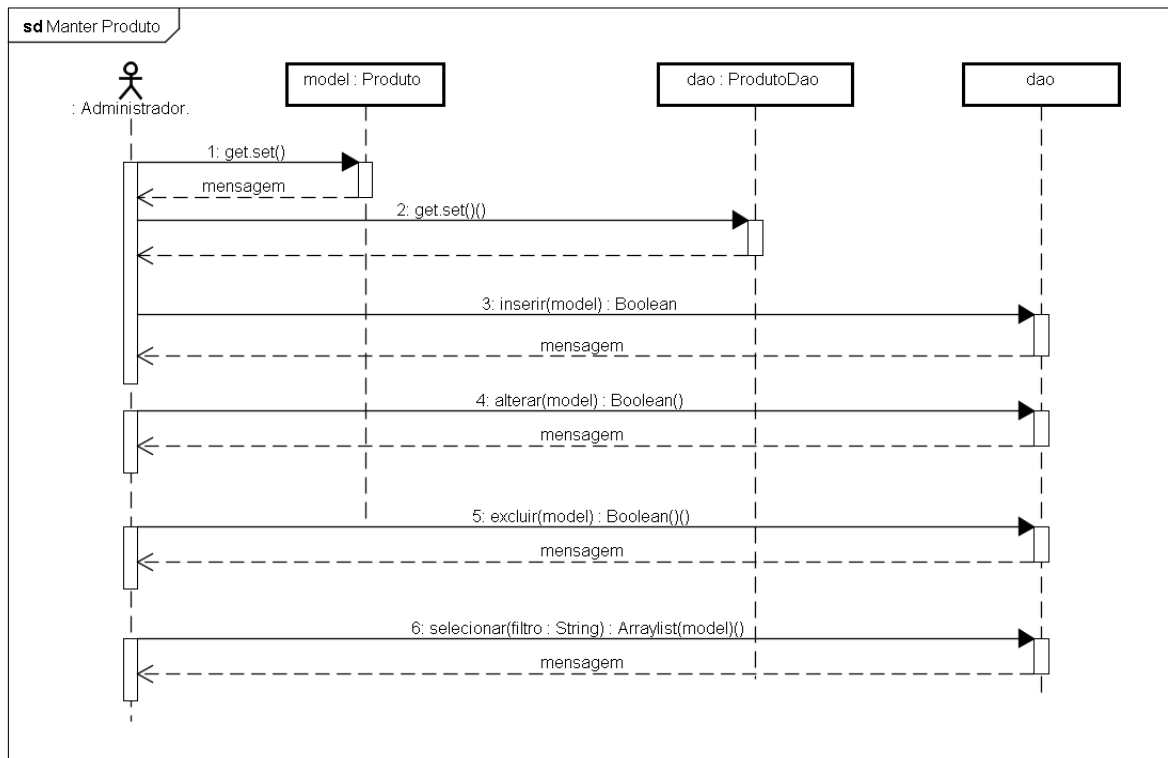


Figura 25: Diagrama de Sequência - Manter Produto

### 3.7.2.5.1. Manter Produto – Use Case 7

A seguir, é demonstrado através da figura 26, o caso de uso Manter Produto, que terá seu acesso pelo administrador do sistema, e irá cadastrar uma vasta quantidade de produtos, para que os administradores dos supermercados escolham quais oferta disponibilizaram e qual será o preço, com isso temos que ter praticamente todos os produtos que vendem nos supermercado cadastrado no sistema.

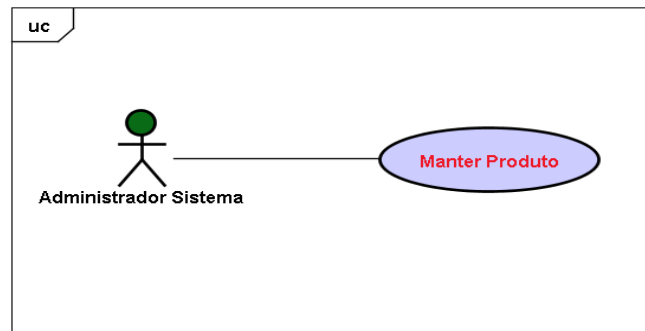


Figura 26: Manter Produto – Use Case 7

Tabela 7: Especificação de Use Case 7 - Manter Produto

<b>Nome da Use Case 7</b>	Manter Produto
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	<p>1- O sistema solicita os dados necessários para cadastrar o produto.</p> <p>2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos.</p> <p>3- O sistema solicita os dados para o cadastro da função.</p> <p>4- O administrador sistema informa os dados necessários.</p> <p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem “Produto cadastrado com sucesso”.</p> <p>7- O sistema cadastra o produto.</p>
<b>Cenário Alternativo</b>	Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que o

	<p>produto não está cadastrado.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.6. Diagrama de Sequência - Manter Supermercado Conveniado

Neste caso de Uso, o Administrador do Sistema irá acessar o módulo Supermercados, para poder acrescentar, alterar, excluir e visualizar todos os supermercados conveniados ao sistema.

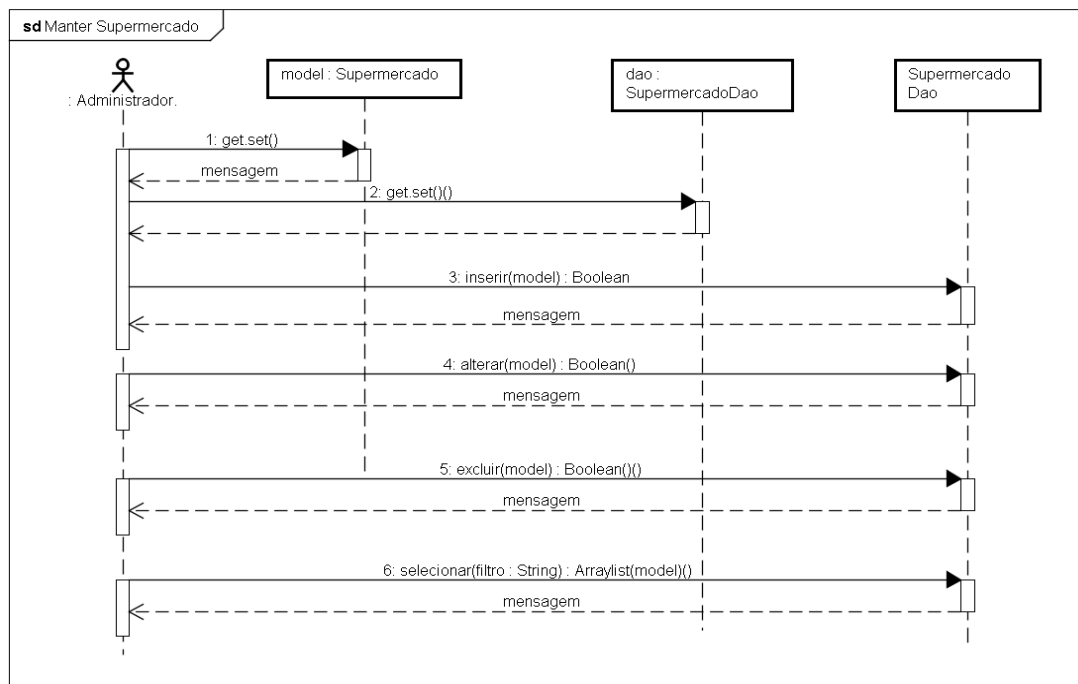


Figura 27: Diagrama de Sequência - Manter Supermercado Conveniado



### 3.7.2.6.1. Manter Supermercado Conveniado – Use Case 8

Este Caso de uso, será responsável pelo cadastro dos Supermercados conveniados ao sistema, os quais, irão disponibilizar as suas ofertas, para a conclusão do cadastro, será necessário escolher um Administrador do supermercado que foi previamente cadastrado no caso de uso “Manter Administrador de Supermercado”.

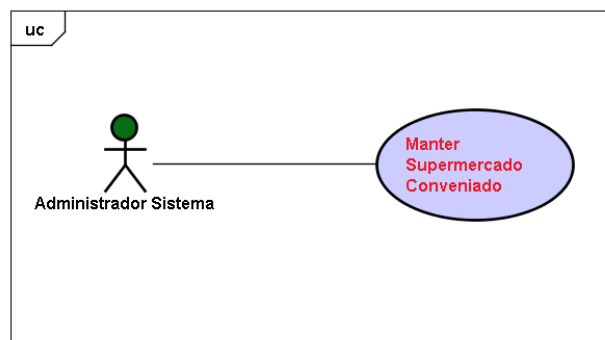


Figura 28: Manter Supermercado Conveniado – Use Case 8

Tabela 8 - Especificação de Use Case 8 - Manter Supermercado Conveniado

<b>Nome da Use Case 8</b>	Manter Supermercado Conveniado
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	1- O sistema solicita os dados necessários para cadastrar o supermercado conveniado. 2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos. 3- O sistema solicita os dados para o cadastro da função. 4- O administrador sistema informa os dados

	<p>necessários.</p> <p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem “Supermercado conveniado cadastrado com sucesso”.</p> <p>7- O sistema cadastra o supermercado conveniado.</p>
<b>Cenário Alternativo</b>	<p>Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que o supermercado conveniado não está cadastrado.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.7. Diagrama de Sequência - Manter Fornecedor

Neste caso de Uso o Administrador do Sistema irá acessar o módulo de Fornecedores, para acrescentar, alterar, excluir e visualizar todos os fornecedores. Essa base de dados irá ser usada para o cadastro de produtos, aonde cada produto tem um fornecedor.

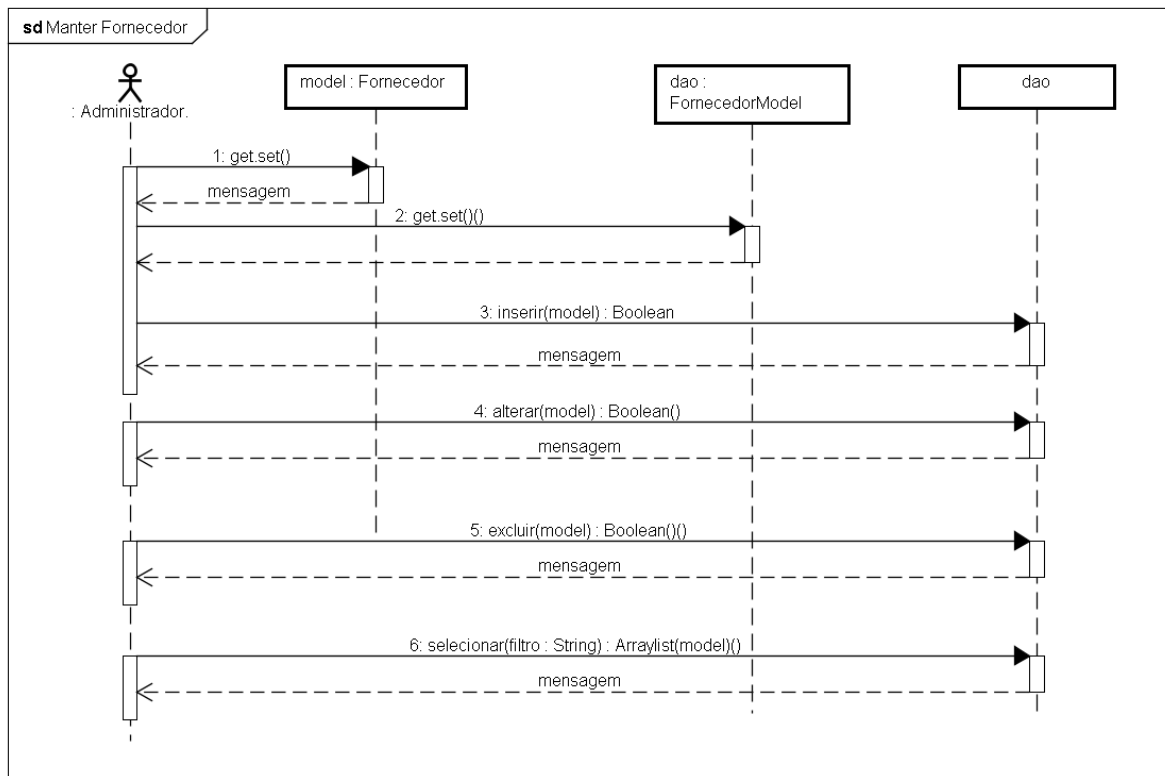


Figura 29: Diagrama de Sequência - Manter Fornecedor

### 3.7.2.7.1. Manter Fornecedor - Use Case 9

A figura 30 ilustra o caso de uso Manter Fornecedor, onde será cadastrado os fornecedores, os quais serão escolhido posteriormente no cadastro de produtos, o seu cadastro é feito pelo administrador do sistema.

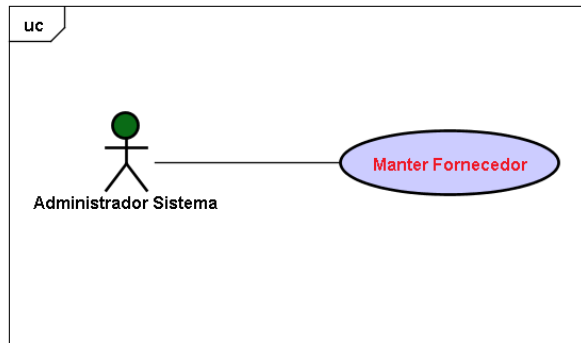


Figura 30: Manter Fornecedor – Use Case 9

Tabela 9: Especificação de Use Case 9 - Manter Fornecedor

<b>Nome da Use Case 9</b>	Manter Fornecedor
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Cenário Principal</b>	<p>1- O sistema solicita os dados necessários para cadastrar o fornecedor.</p> <p>2- O administrador sistema informa os dados de acordo com os campos a serem preenchidos.</p> <p>3- O sistema solicita os dados para o cadastro da função.</p> <p>4- O administrador sistema informa os dados necessários.</p> <p>5- O administrador sistema seleciona a opção Cadastrar</p> <p>6- O sistema emite a mensagem "Fornecedor cadastrado com sucesso".</p> <p>7- O sistema cadastra o fornecedor.</p>
<b>Cenário Alternativo</b>	Se o administrador sistema não informar os dados para o cadastro da função, o sistema informa que o

	<p>fornecedor não está cadastrado.</p> <p>O administrador sistema poderá cancelar o processo durante o cadastro.</p>
<b>Casos de Teste</b>	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

### 3.7.2.8. Diagrama de Sequência - Emitir Relatório de Supermercado

O Administrador do Sistema poderá emitir um relatório de todos os supermercados cadastrados no sistema.

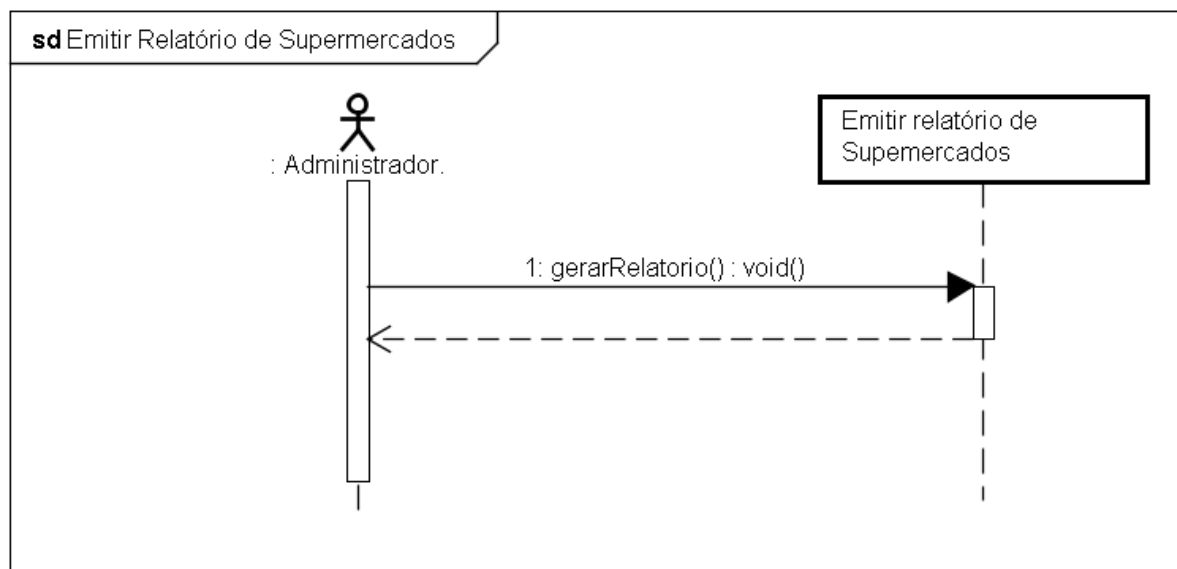


Figura 31: Diagrama de Sequência - Emitir Relatório de Supermercado

### 3.7.2.8.1. Emitir Relatório de Supermercado - Use Case 10

A seguir está a figura 32, onde o administrador do sistema terá um relatório de todos os supermercados cadastrado, podendo visualizar todos seus dados cadastrais e a quantidade de ofertas já criadas por eles.

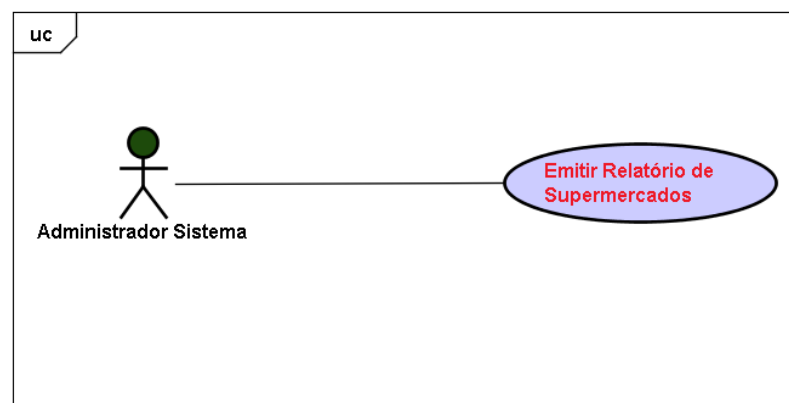


Figura 32: Emitir Relatório de Supermercado – Use Case 10

Tabela 10: Especificação de Use Case 10 - Emitir Relatório de Supermercado

<b>Nome da Use Case 10</b>	Emitir Relatório de Supermercado
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Descrição</b>	Os relatórios serão impressos conforme solicitados.
	1- O sistema disponibiliza os dados necessários para o relatório. 2- O administrador sistema seleciona o botão Visualizar Relatório. 3- O administrador sistema seleciona o botão Imprimir.

<b>Cenário Principal</b>	4- O sistema imprime o relatório com sucesso.
<b>Cenário Alternativo</b>	O administrador sistema poderá visualizar o relatório e não imprimir.
<b>Casos de Teste</b>	4.1- O administrador sistema cancela a operação.

### 3.7.2.9. Diagrama de Sequência - Emitir Relatório de Fornecedor

O Administrador do Sistema poderá emitir um relatório de todos os fornecedores cadastrados no sistema.

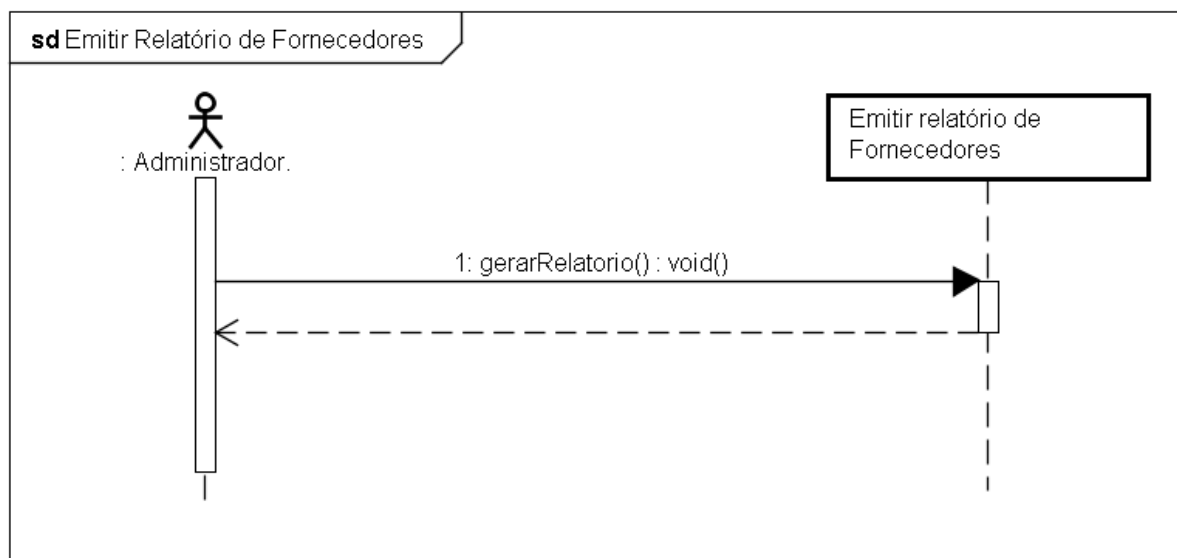


Figura 33: Diagrama de Sequência - Emitir Relatório de Fornecedor

#### 3.7.2.9.1. Emitir Relatório de Fornecedor - Use Case 11

Na figura a seguir é demonstrado o Relatório de Fornecedores, onde o administrador do sistema pode visualizar todos os fornecedores cadastrados e os seus respectivos produtos associados.

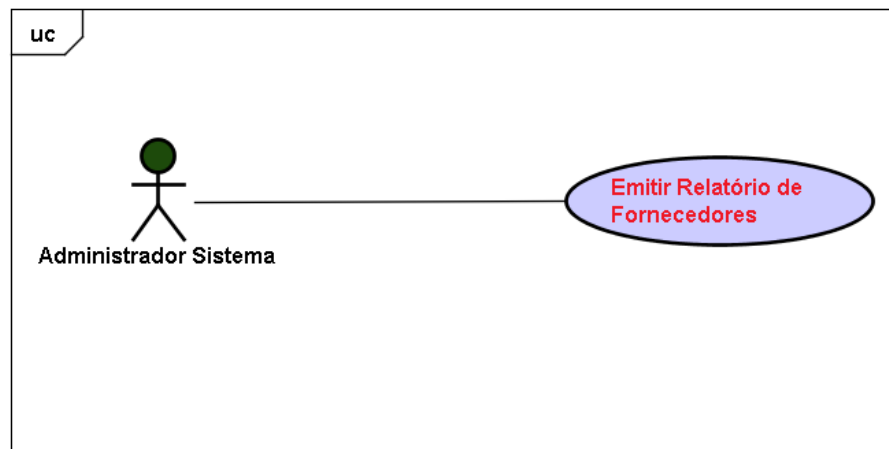


Figura 34: Emitir Relatório de Fornecedor – Use Case 11

Tabela 11: Especificação de *Use Case 11* - Emitir Relatório de Fornecedor

<b>Nome da Use Case 11</b>	Emitir Relatório de Fornecedor
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Descrição</b>	Os relatórios serão impressos conforme solicitados.
<b>Cenário Principal</b>	1- O sistema disponibiliza os dados necessários para o relatório. 2- O administrador sistema seleciona o botão Visualizar Relatório. 3- O administrador sistema seleciona o botão Imprimir. 4- O sistema imprime o relatório com sucesso.
<b>Cenário Alternativo</b>	O administrador sistema poderá visualizar o relatório e não imprimir.
<b>Casos de Teste</b>	4.1- O administrador sistema cancela a operação.



### 3.7.2.10. Diagrama de Sequência - Emitir Relatório de Produto

O Administrador do Sistema poderá emitir um relatório de todos os produtos cadastrados no sistema.

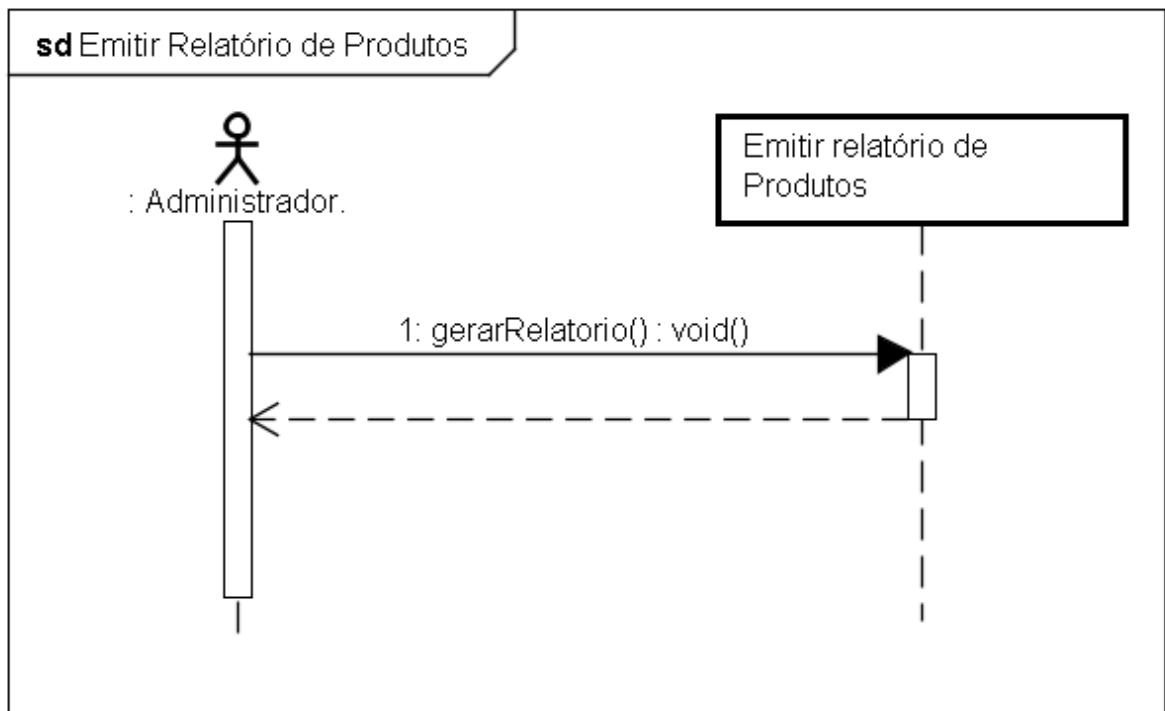


Figura 35: Diagrama de Sequência - Emitir Relatório de Produto

#### 3.7.2.10.1. Emitir Relatório de Produto – Use Case 12

A figura 36 ilustra o relatório de produto, onde através dele, poderá ver todo os produtos cadastrados no sistema.

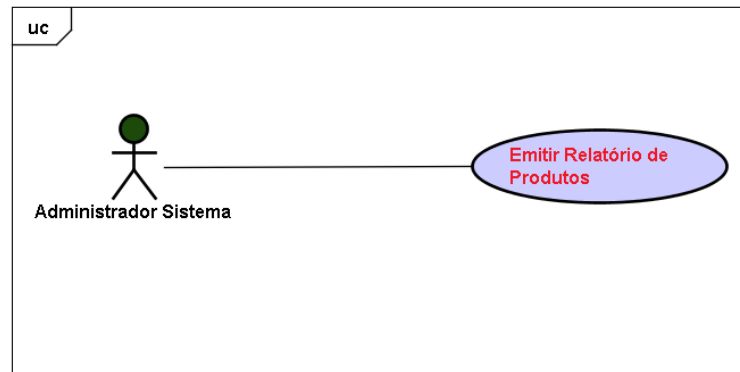


Figura 36: Emitir Relatório de Produto – Use Case 12

Tabela 12: Especificação de Use Case 12 - Emitir Relatório de Produto

<b>Nome da Use Case 12</b>	Emitir Relatório de Produto
<b>Ator (es)</b>	Administrador Sistema
<b>Pré-Condição</b>	O administrador sistema deverá estar autenticado no sistema.
<b>Descrição</b>	Os relatórios serão impressos conforme solicitados.
<b>Cenário Principal</b>	1- O sistema disponibiliza os dados necessários para o relatório. 2- O administrador sistema seleciona o botão Visualizar Relatório. 3- O administrador sistema seleciona o botão Imprimir. 4- O sistema imprime o relatório com sucesso.
<b>Cenário Alternativo</b>	O administrador sistema poderá visualizar o relatório e não imprimir.
<b>Casos de Teste</b>	4.1- O administrador sistema cancela a operação.

### 3.7.3. Especificações do usuário Android

### 3.7.3.1. Diagrama de Sequência - Buscar Produto

O usuário *android* acessará o módulo de buscar produtos, onde terá o acesso através dele a suas ofertas.

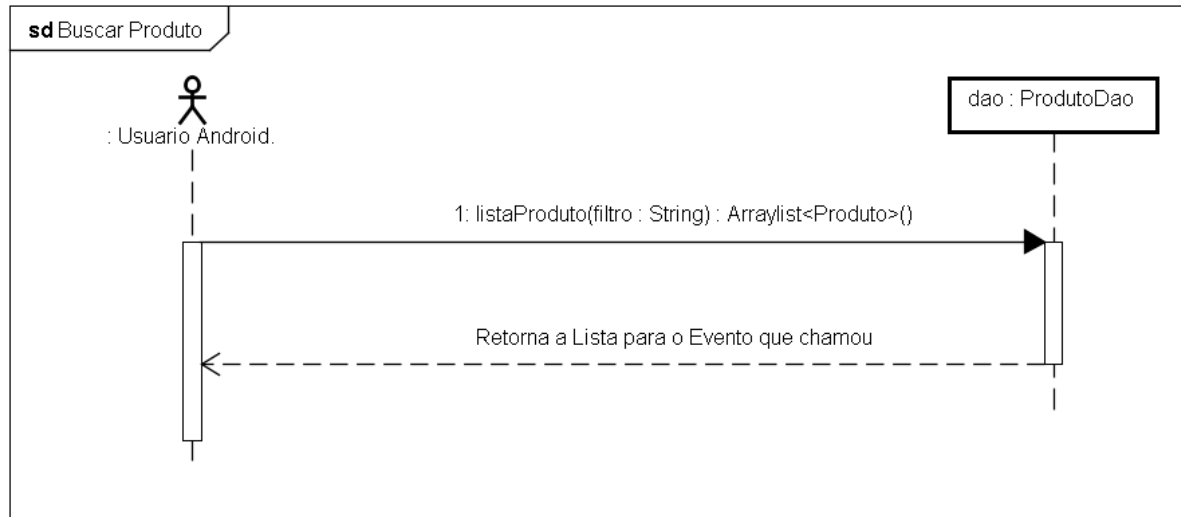


Figura 37: Diagrama de Sequência - Buscar Produto

#### 3.7.3.1.1. Buscar Produto - Use Case 13

Nesse módulo, o usuário poderá buscar qualquer produto cadastrado no sistema e retornarão pra ele somente as ofertas associadas a esse produto e os supermercados correspondentes a elas.

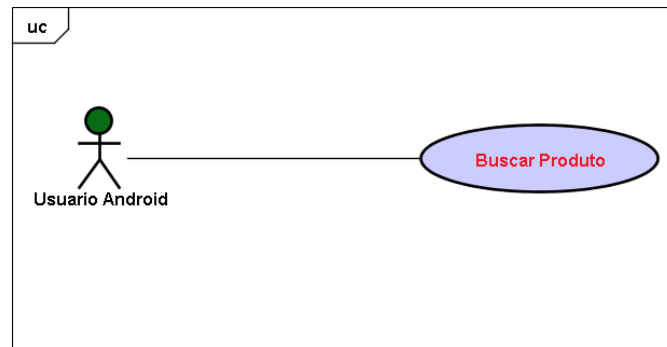


Figura 38: Buscar Produto – Use Case 13

Tabela 13: Especificação de Use Case 13 - Buscar Produto

<b>Nome da Use Case 13</b>	Buscar Produto
<b>Ator (es)</b>	Usuário <i>Android</i>
<b>Pré-Condição</b>	Ter o aplicativo <i>Android</i> instalado no celular.
<b>Cenário Principal</b>	1- O aplicativo solicita os dados necessários para buscar o produto. 2- O usuário <i>Android</i> informa os dados de acordo com a categoria escolhida. 3- O usuário <i>Android</i> informa os dados necessários.
<b>Cenário Alternativo</b>	O usuário <i>Android</i> poderá cancelar o processo durante a busca.

### 3.7.3.2. Diagrama de Sequência - Realizar Compra

Nesse módulo o usuário escolherá a lista de compras feitas, definir qual mercado em que está e realizar sua compra, pra futuramente poder acompanhar os gastos com supermercado.

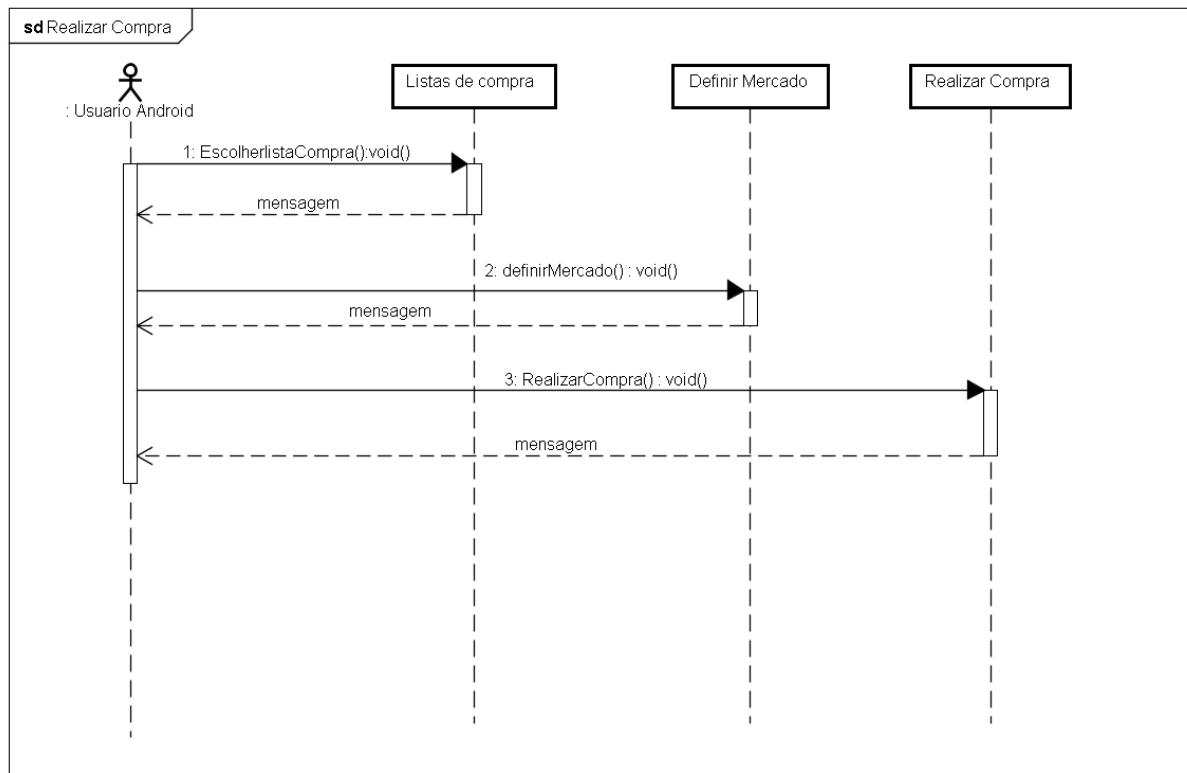


Figura 39: Diagrama de Sequência - Realizar Compra

### 3.7.3.2.1. Realizar Compra - Use Case 14

Na figura 40, o usuário *Android* realizará a sua compra, ou partindo do zero, adicionando produtos assim que for pegando nas prateleiras ou escolher uma lista de compra pré realizada com os produtos que necessita comprar, e definindo um supermercado para depois ter um controle de quanto gasta em cada supermercado.

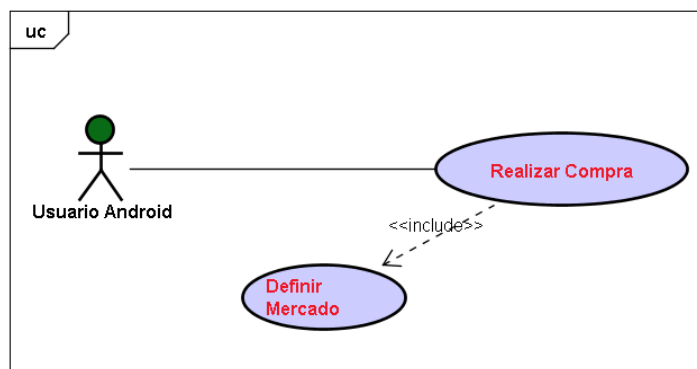


Figura 40: Buscar Produto – Use Case 14

Tabela 14: Especificação de Use Case 14 – Realizar Compra

<b>Nome da Use Case 14</b>	Realizar Compra
<b>Ator (es)</b>	Usuário <i>Android</i>
<b>Pré-Condição</b>	Ter o aplicativo <i>Android</i> instalado no celular.
<b>Cenário Principal</b>	1- O aplicativo solicita os dados necessários para realizar compra. 2- O usuário <i>android</i> poderá executar uma compra usando uma lista de. compras ou escolhendo um produto no momento 3- O usuário <i>android</i> informa os dados necessários. 4- Após ter feito a sua compra, ela é finalizada.
<b>Cenário Alternativo</b>	O usuário <i>android</i> poderá cancelar o processo durante a compra.

### 3.7.3.3. Diagrama de Sequência - Acompanhar Gastos

O usuário poderá acompanhar todos os gastos com supermercado, desde o gasto total, como o valor gasto em cada um.

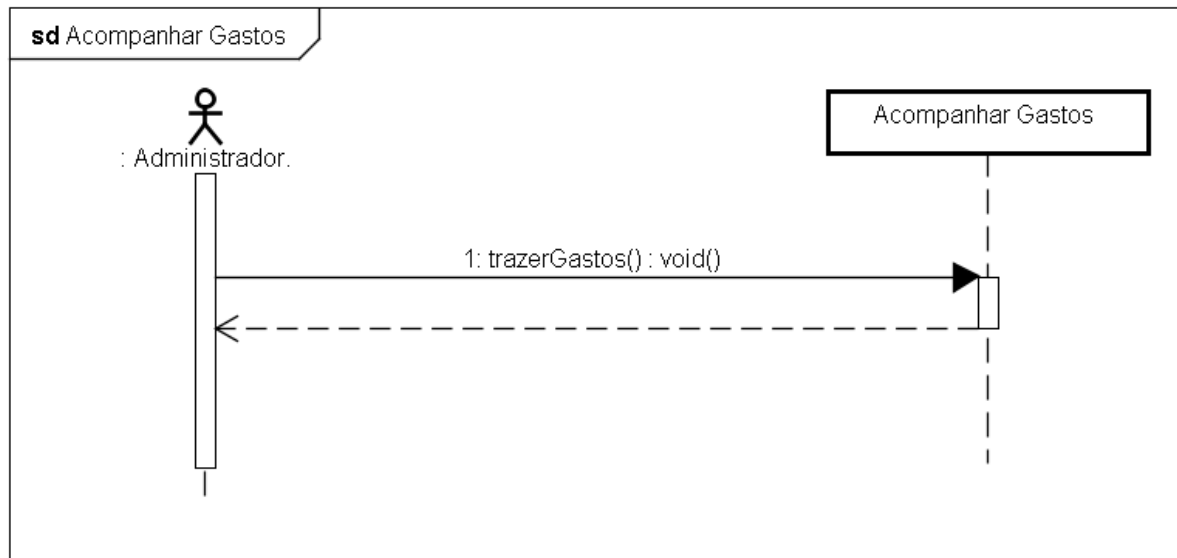


Figura 41: Diagrama de Sequência - Acompanhar Gastos

### 3.7.3.3.1. Acompanhar Gastos – Use Case 15

Através deste Caso de Uso, o usuário *Android* poderá acompanhar todos os seus gastos referente as suas compras, podendo filtrar por supermercado, como listar todos, podendo filtrar por datas.

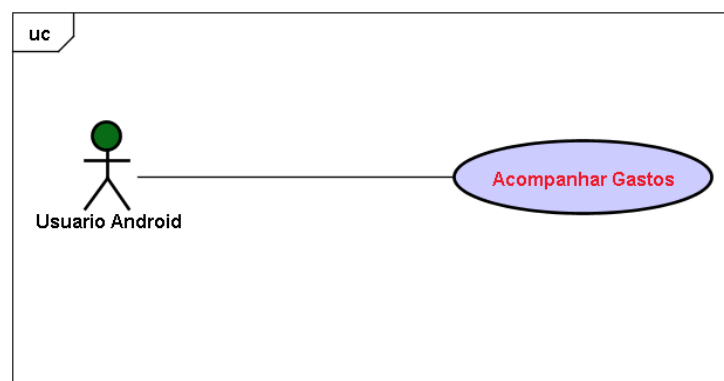


Figura 42: Acompanhar Gastos – Use Case 15

Tabela 15: Especificação de *Use Case* 15 - Acompanhar Gastos

<b>Nome da Use Case 15</b>	Acompanhar Gastos
<b>Ator (es)</b>	Usuário <i>Android</i>
<b>Pré-Condição</b>	Ter o aplicativo <i>Android</i> instalado no celular.
<b>Cenário Principal</b>	1- O aplicativo solicita os dados necessários para acompanhar gastos. 2- O usuário <i>android</i> informa os dados de acordo com o tipo de relatório pedido. 3- O usuário <i>android</i> informa os dados necessários. 4- O relatório estará disponível.
<b>Cenário Alternativo</b>	O usuário <i>android</i> poderá cancelar o processo durante a busca.

### 3.7.3.4. Diagrama de Sequência - Emitir Relatório Lista de Compra

Aqui será possível ver todas as listas de compras e seus devidos produtos em forma de um relatório.

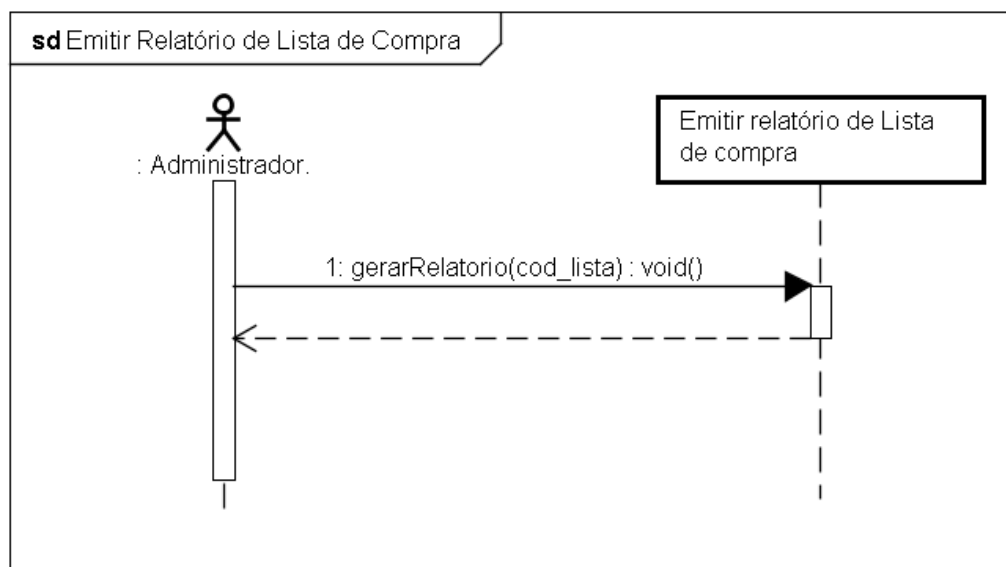


Figura 43: Diagrama de Sequência - Emitir Relatório Lista de Compra



### 3.7.3.4.1. Emitir Relatório Lista de Compra - Use Case 16

Na figura 14, o usuário poderá visualizar um relatório de todas as listas de compras criadas por ele e seus respectivos produtos.

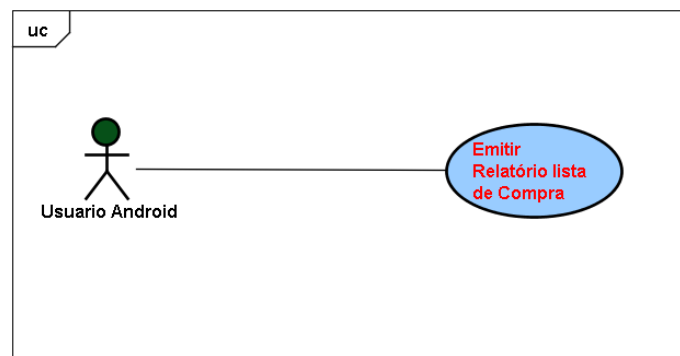


Figura 44: Emitir Relatório Lista de Compra – Use Case 16

Tabela 16: Especificação de Use Case 16 - Emitir Relatório Lista de Compra

<b>Nome da Use Case 16</b>	Emitir Relatório Lista de Compra
<b>Ator (es)</b>	Usuário <i>Android</i>
<b>Pré-Condição</b>	Ter o aplicativo <i>Android</i> instalado no celular.
<b>Cenário Principal</b>	1- O aplicativo solicita os dados necessários para emitir relatório lista de compra. 2- O usuário <i>android</i> informa os dados de acordo com o relatório pedido. 3- O usuário <i>android</i> informa os dados necessários. 4- O relatório estará disponível
<b>Cenário Alternativo</b>	O usuário <i>android</i> poderá cancelar o processo durante a busca.

### 3.7.3.5. Diagrama de Sequência - Manter Lista de Compra

Nesse módulo o usuário poderá criar sua lista de compra, alterar, excluir, definir os produtos que irão constar em cada lista, para posteriormente usar em algum determinado supermercado, e ter o controle de quanto gasta no supermercado escolhido.

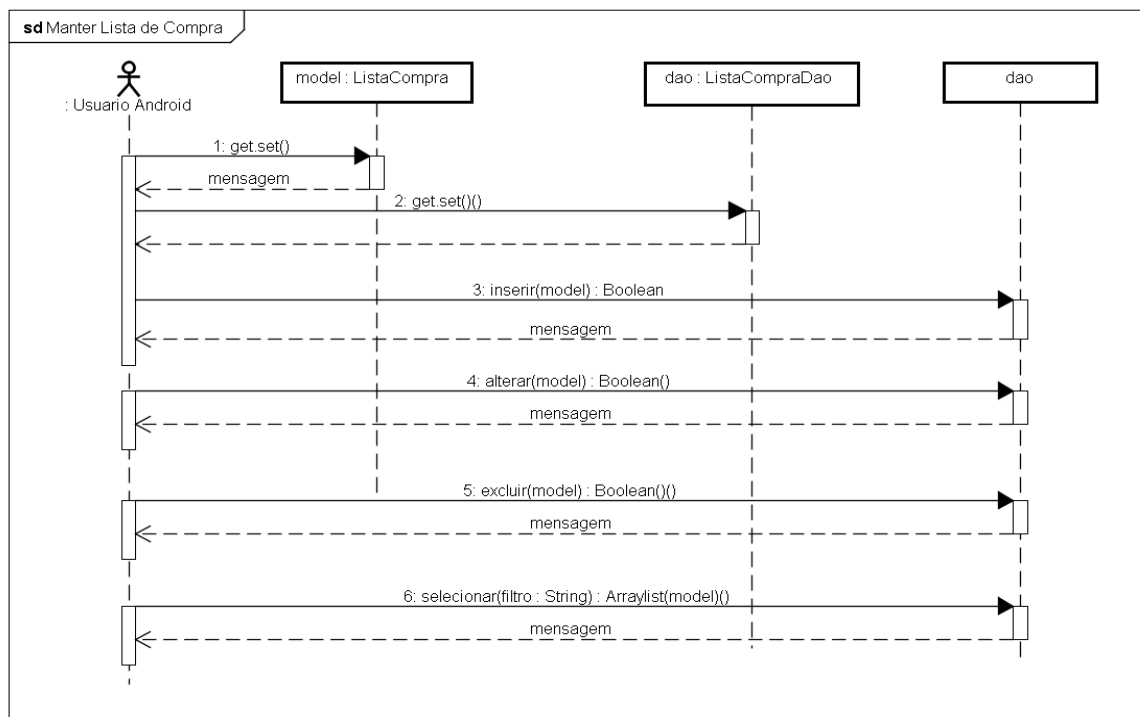


Figura 45: Diagrama de Sequência - Manter Lista de Compra

#### 3.7.3.5.1. Manter Lista de Compra – Use Case 17

Na figura 46, é demonstrado o caso de uso Manter Lista de Compra, que será utilizado pelo Usuário do *Android*, ele escolherá os produtos baseado no cadastro de produtos e assim que chegar em algum supermercado, realizará a sua compra usando esta lista.

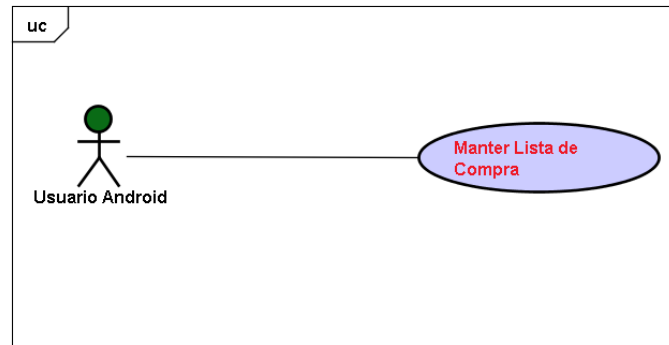


Figura 46: Manter Lista de Compra – Use Case 17

Tabela 17: Especificação de Use Case 17 - Manter Lista de Compra

<b>Nome da Use Case 17</b>	Cadastrar Lista de Compra
<b>Ator (es)</b>	Usuário <i>Android</i>
<b>Pré-Condição</b>	Ter o aplicativo <i>Android</i> instalado no celular.
<b>Cenário Principal</b>	1- O aplicativo solicita os dados necessários para cadastrar lista de compra. 2- O usuário <i>android</i> informa os dados necessários para o cadastro. 3- O usuário <i>android</i> cadastra o produto na lista de compras.
<b>Cenário Alternativo</b>	O usuário <i>android</i> poderá cancelar o processo durante o cadastro.

### 3.8. Diagramas de Atividades

#### 3.8.1. Diagrama de Atividade da Aplicação Android

Na figura 47, é demonstrado o diagrama de Atividade referente à Aplicação *Android*, começando pelo Menu o usuário tem Acesso a vários módulos, dentre eles a busca de produto, lista de compras, realizar compra, acompanhar gastos e configurações. A Partir do módulo acessado ele fará suas ações e poderá cancelar sua ação retornando ao Menu ou concluí-la com êxito.

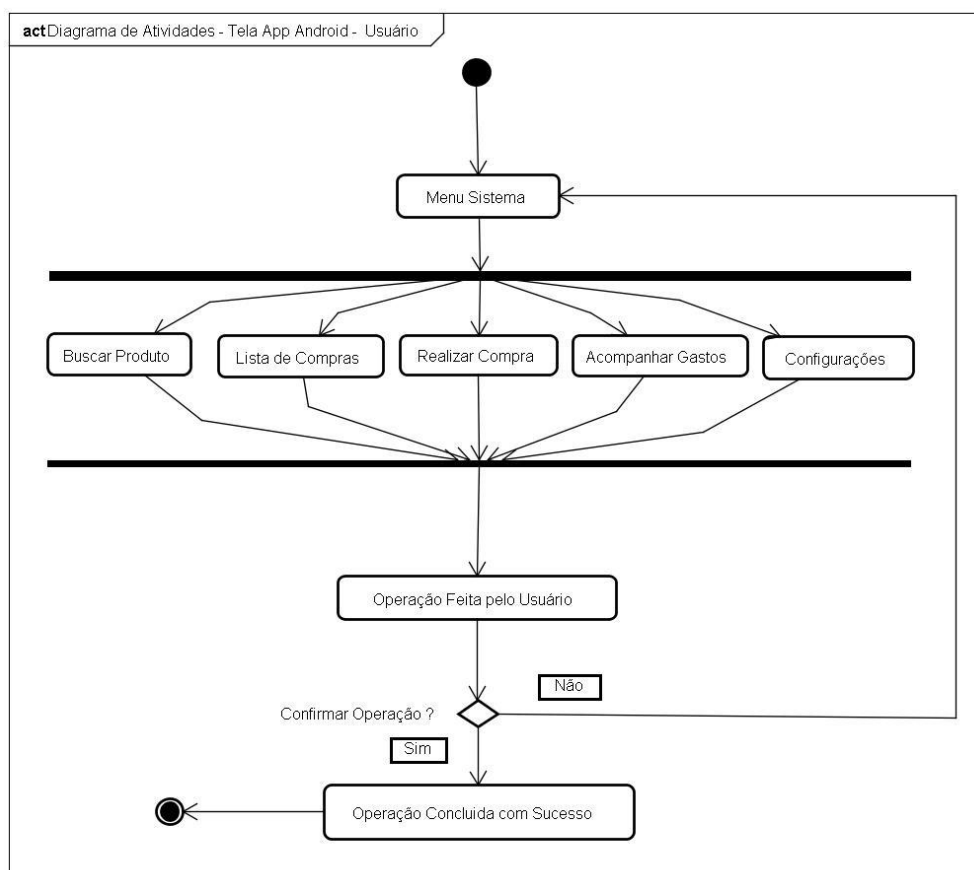


Figura 47: Diagrama de Atividade - Aplicação *Android*

### 3.8.2. Diagrama de Atividade do Website referente ao Supermercado

Na figura 48, é demonstrado o diagrama de Atividade referente ao Website, na visão do Administrador do supermercado, onde quem irá acessar será somente ele. A partir do momento que entrar com suas credenciais (Login) será visualizado o menu, contendo os módulos Manter Ofertas e Histórico de Ofertas, dos quais são carregados com os dados referentes ao supermercado que o gerente é cadastrado.

No Modulo Manter Ofertas ele irá definir as ofertas, que valerá por um determinado período, já no Histórico ofertas, ele visualizara todas as ofertas que já ocorreu podendo fazer uma impressão das próprias; terminando as ações ele voltará ao Menu.

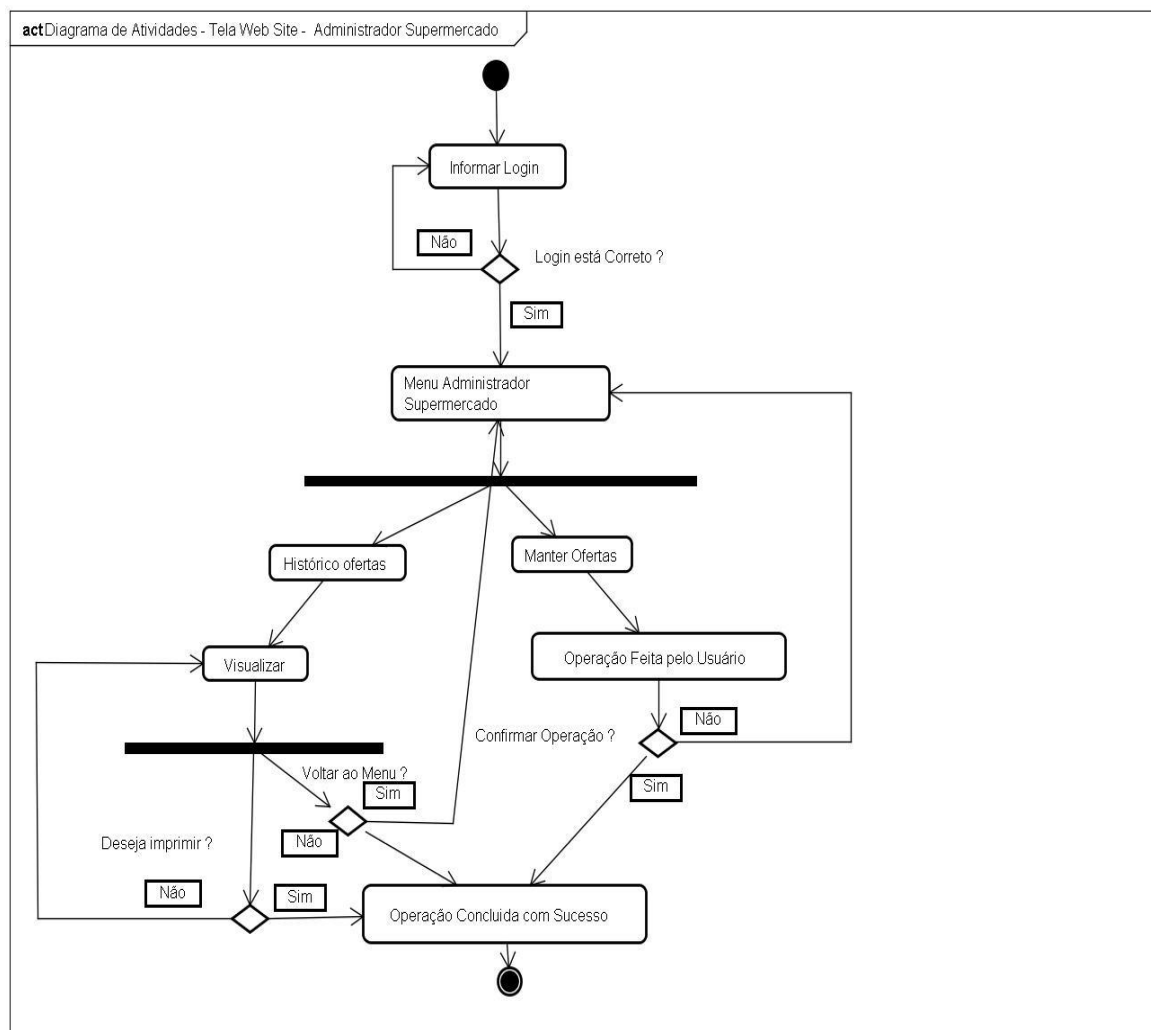


Figura 48: Diagrama de Atividade: Website destinado ao Supermercado

### 3.8.3. Diagrama de Atividade referente ao Administrador

Na figura 49, é demonstrado o diagrama de Atividade referente ao Website, na visão do Administrador do sistema, onde quem irá acessar será somente ele. A partir do momento que entrar com suas credenciais (Login) será visualizado o menu, contendo os módulos Manter Fornecedores, Manter Produtos, Manter bairro, Manter Rua, Manter Supermercados, Manter Administradores Supermercados.

A Partir do módulo acessado ele fará suas ações e poderá cancelar sua ação retornando ao Menu ou concluí-la com êxito.

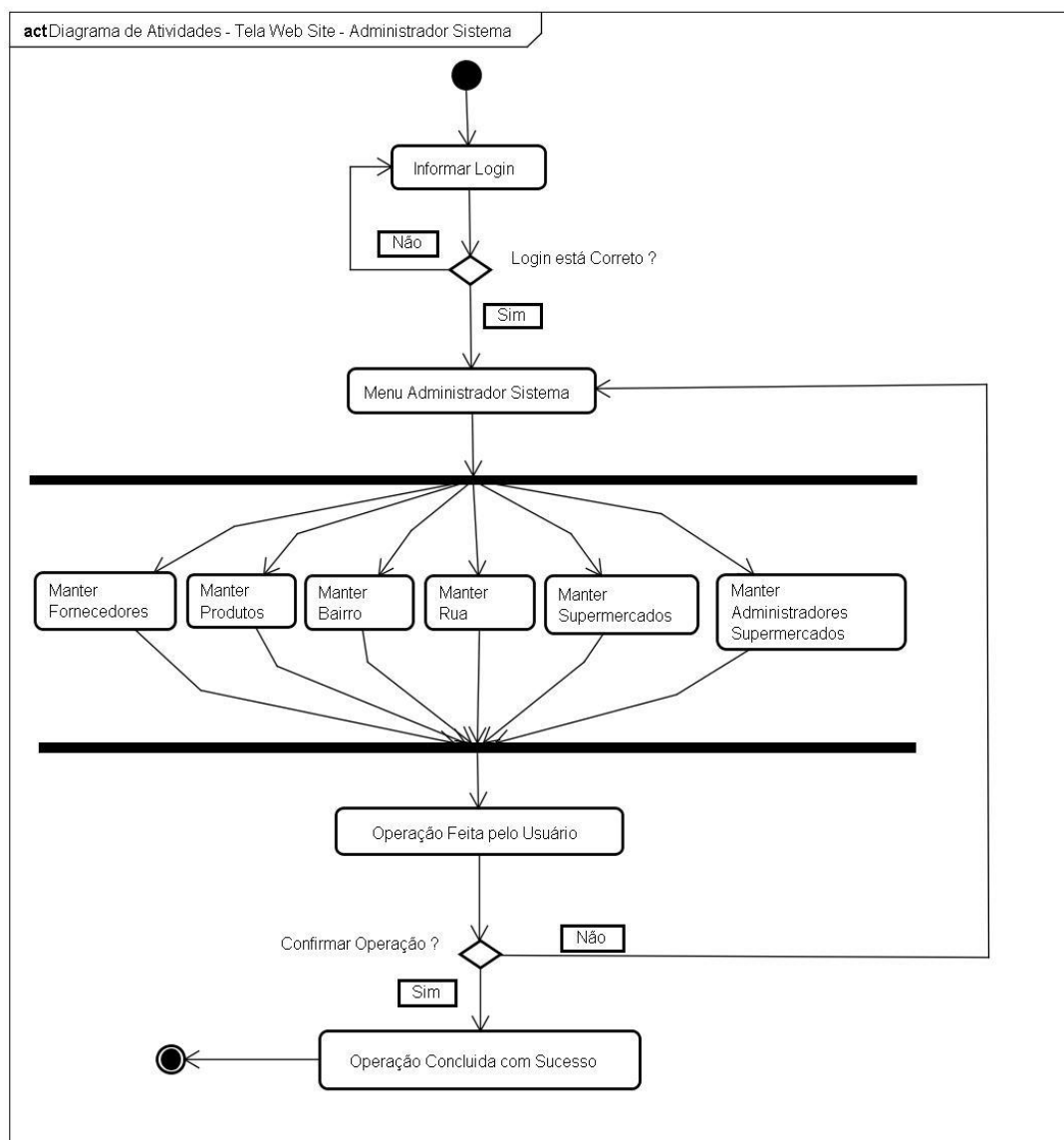


Figura 49: Diagrama de Atividade – Website destinado ao Administrador

#### 4. Conclusão

Com o software CompareOffer, o cliente terá em suas mãos um “*folhetim eletrônico*”, onde não precisará mais levar vários panfletos de supermercado cada vez que for fazer uma compra, pois com o programa em tempo real, ele poderá verificar o preço dos produtos em outros supermercados. Além da ferramenta servir para os próprios supermercados verem o preço das ofertas dos concorrentes.

Foram adquiridos grandes conhecimentos com o desenvolvimento do projeto, por ser o primeiro desenvolvido em *Android*, ocorreu várias dificuldades, não encontradas em sistemas Desktops, pois sua arquitetura de desenvolvimento é totalmente diferente, e com isso veio à necessidade de aprender também outras tecnologias que iriam trabalhar em paralelo para o total funcionamento do sistema, como o WebService, conhecido por mim, mas nunca até então desenvolvido.

Como não é um software requerido por uma empresa, na questão de levantar requisitos, não há como esperar de momento que supre as necessidades de um empresa, pois é uma ferramenta criada para eles (clientes, e supermercados). Foi pensada para que engloba a ideia de obtenção e comparação de preços, diferentes das outras ferramentas que são feitas modeladas para as próprias.

O esperado com o desenvolvimento deste trabalho é a obtenção do sucesso deste software, de forma que os usuários, tanto os supermercadistas como os consumidores deste ramo, possam aderir à utilização do mesmo. Assim, chegar ao objetivo deste software ser rentável. Outro objetivo que se espera, é que com esse trabalho outros alunos possam utiliza-lo como base para desenvolvimentos

Acredita-se, que este software irá complementar as formas de marketing já existentes nas empresas, e quem sabe num futuro próximo, ajudar na diminuição de panfletagem indiscriminada e poluidora.

## Referências Bibliográficas

Cura, H. A *História do Android*. Disponível em: <<http://pplware.sapo.pt/google/a-istoria-do-android/>>. Acesso em: 15 de maio de 2012.

Deitel, H. M.; Deitel P. J. *Java como programar*. Trad. Lisboa, Carlos Arthur Lang,. 4.ed. Porto Alegre, RS : BOOKMAN Companhia Editora, 2003.

Freitas, Welbson (2007). *História do Java*. Disponível em: <<http://infoblogs.com.br/frame/goframe.action?contentId=13593>>. Acesso em: 13 de junho de 2012.

Garattoni, B. (2011). *Donos de Android comem pizza, ganham menos e votam na direita*. Disponível em: <<http://super.abril.com.br/blogs/rebit/donos-de-android-comem-pizza-ganham-menos-e-votam-em-partidos-de-direita/>>. Acesso em: 15 de maio de 2012.

Geary, D; Horstmann, C. *Core: JavaServer Faces*. Third Edition Redwood Shores: Prenticce Hall, 2010. 636 p.

JUNIOR, Ivaldir. *Sequenciamento de atividades*. Pernambuco. Disponível em: <<http://www.cin.ufpe.br/~if717/Pmbok2000/>>. Acesso em: 18 de out. de 2012.

Pacievitch Y. *História do Java*. Disponível em: <<http://www.infoescola.com/informatica/historia-do-java/>>. Acesso em: 12 de junho.

Silva, L. A. *Programando Passo a Passo*. Apostila. Ed. 3ª. Disponível em: <[http://www.4shared.com/office/MDcHvJaL/apostila\\_de\\_android\\_-\\_programa.html](http://www.4shared.com/office/MDcHvJaL/apostila_de_android_-_programa.html)>. Acesso em: 20 de maio de 2012.





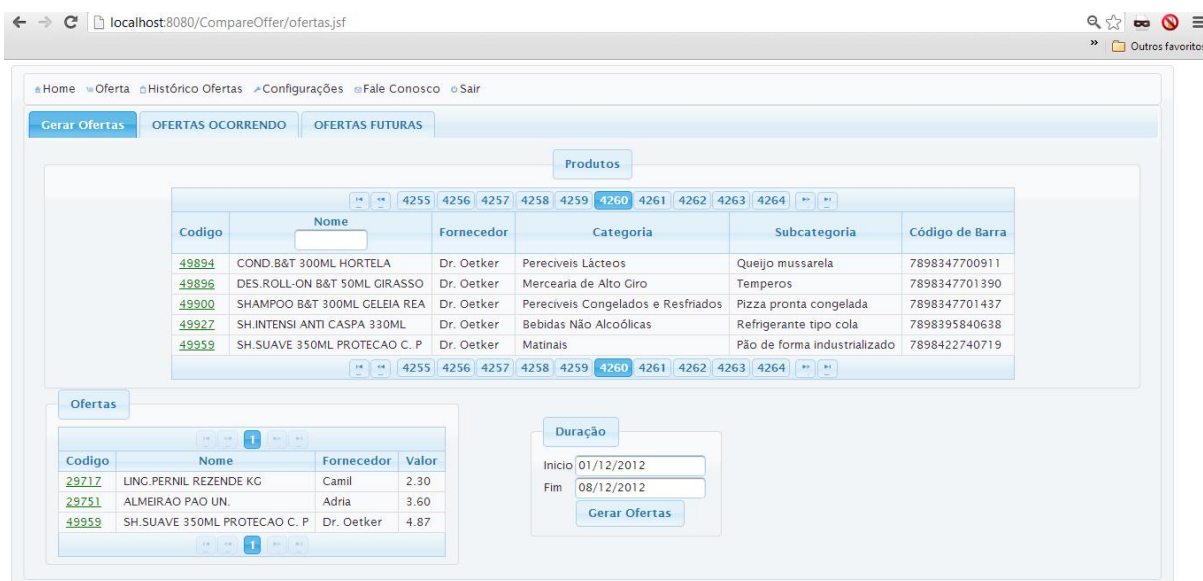
Fundação Educacional do Município de Assis  
Instituto Municipal de Ensino Superior de Assis  
Campos "José Santilli Sobrinho"

Sobre o PostgreSQL. Disponível em: <<http://www.postgresql.org.br/sobre>>. Acesso em: 18 de outubro de 2012.

Torres, A. *O Android é realmente software livre?* Disponível em: <<http://cibermundi.wordpress.com/2012/03/03/o-android-e-realmente-software-livre/>>. Acesso em: 20 de maio de 2012.

## Anexo I

Na Figura 1 é demonstrado a Pagina de gerenciamento de Oferta, onde o gerente do supermercado escolhe o produto, define o valor, escolhe o Início e o Fim e clica em Gerar.



localhost:8080/CompareOffer/ofertas.jsf

Home Oferta Histórico Ofertas Configurações Fale Conosco Sair

Gerar Ofertas OFERTAS OCORRENDO OFERTAS FUTURAS

Produtos

Código	Nome	Fornecedor	Categoria	Subcategoria	Código de Barra
49894	COND.B&T 300ML HORTELA	Dr. Oetker	Perecíveis Lácteos	Queijo mussarela	7898347700911
49896	DES.ROLL-ON B&T 50ML GIRASSO	Dr. Oetker	Mercearia de Alto Giro	Temperos	7898347701390
49900	SHAMPOO B&T 300ML GELEIA REA	Dr. Oetker	Perecíveis Congelados e Resfriados	Pizza pronta congelada	7898347701437
49927	SH INTENSI ANTI CASPA 330ML	Dr. Oetker	Bebidas Não Alcoólicas	Refrigerante tipo cola	7898395840638
49959	SH.SUAVE 350ML PROTECAO C. P	Dr. Oetker	Matinais	Pão de forma industrializado	7898422740719

Ofertas

Código	Nome	Fornecedor	Valor
29717	LING.PERNIL REZENDE KG	Camil	2.30
29751	ALMEIRAO PAO UN.	Adria	3.60
49959	SH.SUAVE 350ML PROTECAO C. P	Dr. Oetker	4.87

Duração

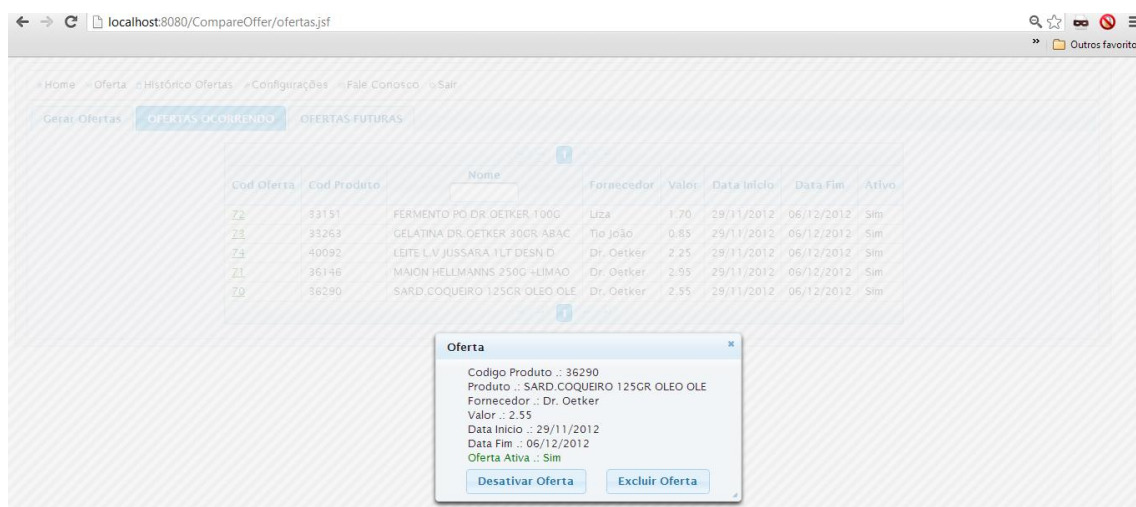
Início 01/12/2012

Fim 08/12/2012

Gerar Ofertas

Figura 1: Gerar Ofertas

Na Figura 2 é demonstrado as Ofertas que já estão ocorrendo, e clicando em cima delas será aberto um Pop-up com a Possibilidade de Desabilitar ou Habilitar uma Oferta de acordo com o seu estado.



Cod Oferta	Cod Produto	Nome	Fornecedor	Valor	Data Inicio	Data Fim	Ativo
22	33151	FERMENTO PO DR OETKER 100G	Liza	1.70	29/11/2012	06/12/2012	Sim
23	33263	GELATINA DR OETKER 30GR ABAC	Tio João	0.85	29/11/2012	06/12/2012	Sim
24	40092	LEITE L.V JUSSARA 1LT DESN D	Dr. Oetker	2.25	29/11/2012	06/12/2012	Sim
21	36146	MAION HELLMANN'S 250G +LIMAO	Dr. Oetker	2.95	29/11/2012	06/12/2012	Sim
20	36290	SARD.COQUEIRO 125GR OLEO OLE	Dr. Oetker	2.55	29/11/2012	06/12/2012	Sim

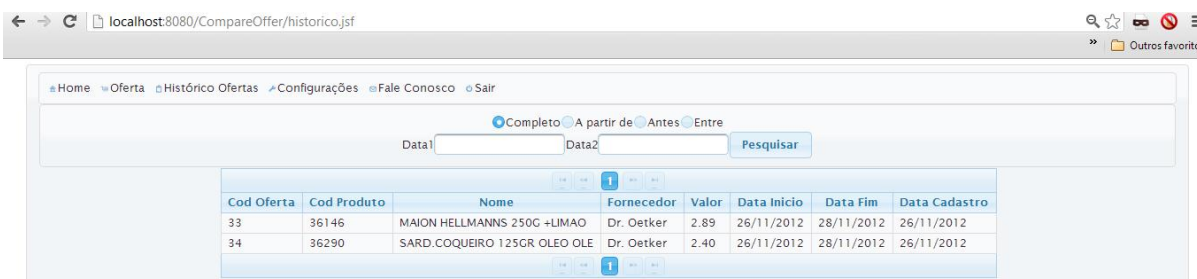
**Oferta**

Codigo Produto : 36290  
 Produto : SARD.COQUEIRO 125GR OLEO OLE  
 Fornecedor : Dr. Oetker  
 Valor : 2.55  
 Data Inicio : 29/11/2012  
 Data Fim : 06/12/2012  
 Oferta Ativa : Sim

Desativar Oferta Excluir Oferta

Figura 2: Ofertas ocorrendo

Na Figura 3 é demonstrado as Ofertas que já ocorreram, podendo ser aplicado vários filtros de datas ou podendo visualizar todas.



Cod Oferta	Cod Produto	Nome	Fornecedor	Valor	Data Inicio	Data Fim	Data Cadastro
33	36146	MAION HELLMANN'S 250G +LIMAO	Dr. Oetker	2.89	26/11/2012	28/11/2012	26/11/2012
34	36290	SARD.COQUEIRO 125GR OLEO OLE	Dr. Oetker	2.40	26/11/2012	28/11/2012	26/11/2012

Figura 3: Histórico Ofertas

Como demonstrado na figura 4 a seguir, o usuário pode se autenticar no aplicativo CompareOffer, se cadastrar ou clicar no About para ler sobre o sistema.



Figura 4: Login

A seguir a tela de About, com o subtítulo do sistema, nome do Desenvolvedor, e do Orientador.

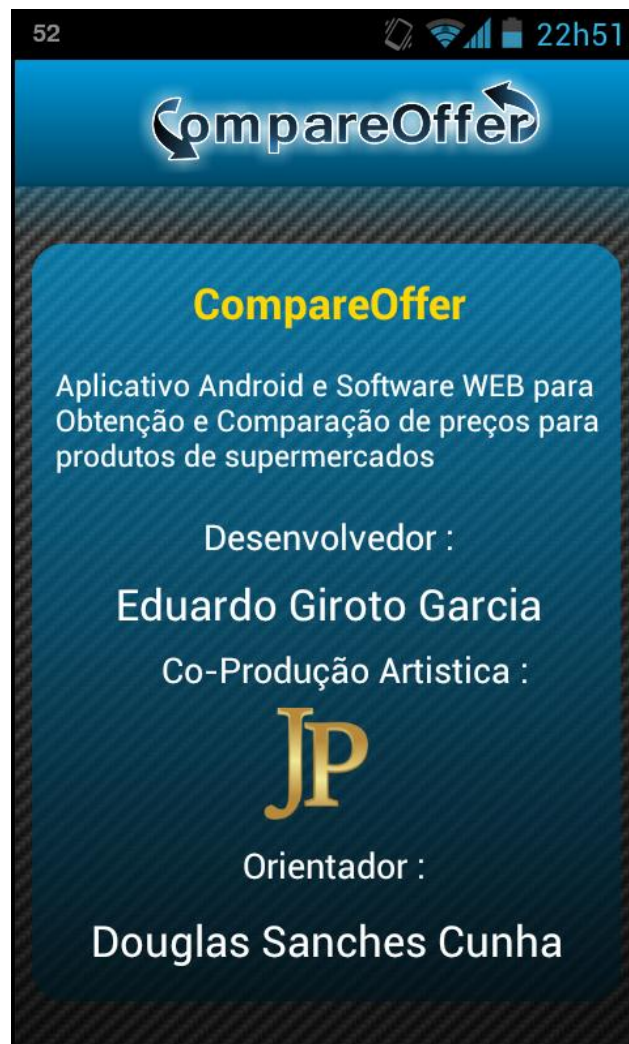


Figura 5: About

Tela Principal, chamada após o usuário se autenticar com êxito, demonstrando o que é cada Ícone do sistema.

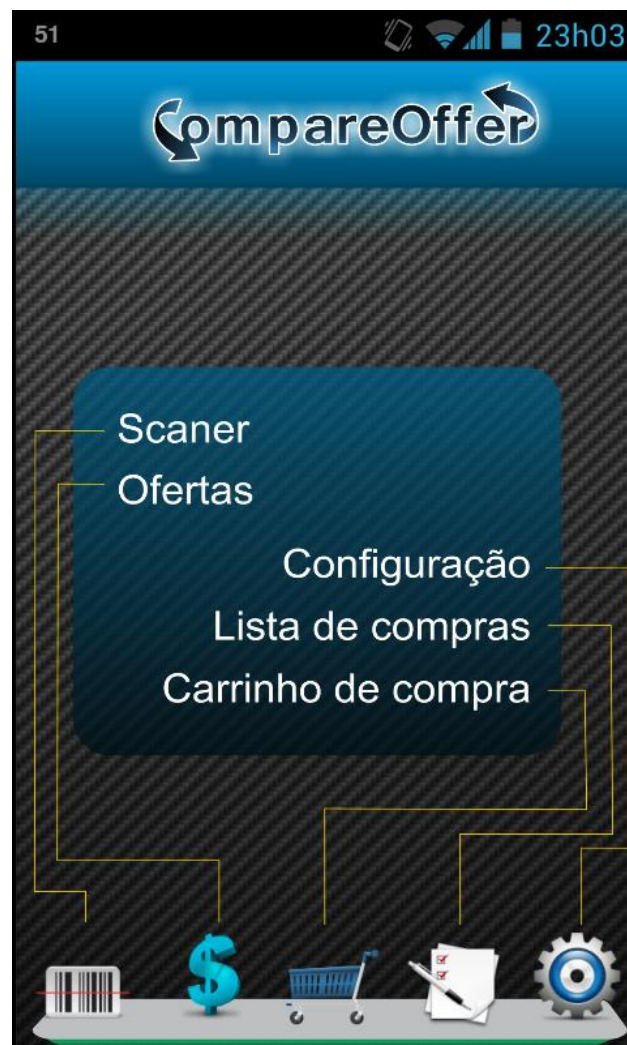


Figura 6: Tela Principal



Na Imagem a Seguir é demonstrada a parte mais importante do sistema, onde consiste em listar todos os mercados aonde encontrasse o produto pesquisado, listando a oferta mais barata acima de todas e clicando na oferta (Figura 8,9) é aberto uma nova janela com o endereço do supermercado onde encontra-se essa oferta.



Figura 7: Lista Oferta

Exemplos de tela, onde mostra o nome completo do produto escolhido, a foto, o valor da oferta, quando ela foi cadastrada e o endereço completo do supermercado.

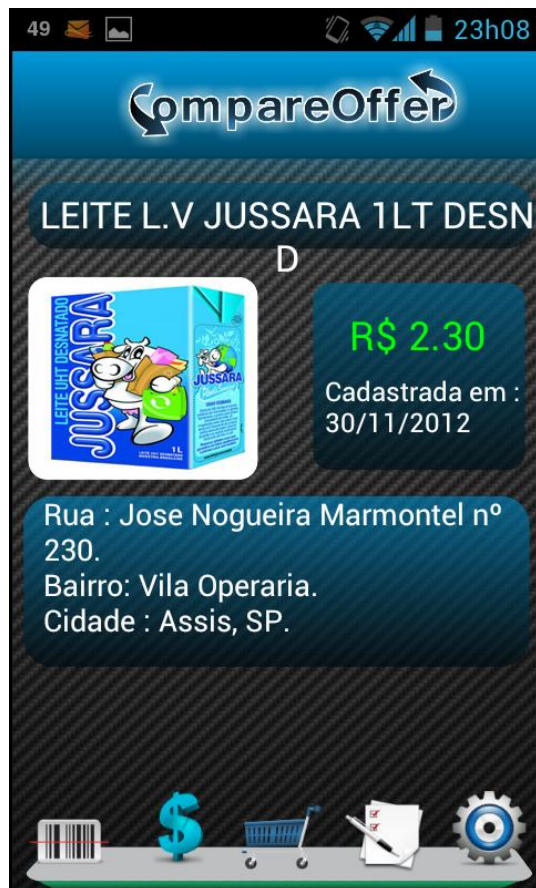


Figura 8: Oferta



Figura 9: Oferta



## Anexo II

### Código do Webservice de Busca de Produto pelo Código de Barra

```
@GET
@Path("/buscarofertascodbarra/{cod_barra}/{token}")
@Produces("application/json; charset=UTF-8")
public String selOfertasCodBarra(@PathParam("cod_barra") String cod_barra, @PathParam("token") String token) {

    AutenticacaoDao aut = new AutenticacaoDao();
    if (aut.ValidarToken(token, "buscarofertas", "cod_barra = " + cod_barra.toString())) {

        OfertaDao oferta = new OfertaDao();
        if(!oferta.AcharCod_Barra(cod_barra)){
            return "Produto não Cadastrado.";
        }

        return new Gson().toJson(oferta.getListaOfertasOcorrendoCodBarra("", cod_barra));
    } else {
        return " !!! Autenticação Invalida no Webservice !!!";
    }
}
```

### Código do Webservice de Busca Imagens através do código de um produto

```
@GET
@Path("/buscarimgofertas/{cod}/{token}")
@Produces("application/json; charset=UTF-8")
public String selImgOfertas(@PathParam("cod") Integer cod, @PathParam("token") String token) {

    AutenticacaoDao aut = new AutenticacaoDao();

    if (aut.ValidarToken(token, "buscarimgofertas", "cod = " + cod.toString())) {

        ImagemDao img = new ImagemDao();

        return new Gson().toJson(img.getListImagens(cod));
    } else {

        return " !!! Autenticação Invalida no Webservice !!!";
    }
}
```

Código do Responsável de conectar no banco e retorna um ArrayList de Ofertas.

```

public ArrayList<OfertaModel> getListaOfertasOcorrendoCodBarra (String filtro,String cod_barra) {
    Connection con = null;
    Statement stmt = null;
    ResultSet rs = null;
    Date data_hoje = Validacoes.DataProgramaParaBanco(Validacoes.dataHoje());

    ArrayList<OfertaModel> lista = new ArrayList<OfertaModel>();
    // iniciando a conexao
    con = Conexao.getInstance().getConnection();
    System.out.println("conectado e preparando listagem");
    try {
        stmt = (Statement) con.createStatement();
        String sql;

        sql = "select supermercado.img, cidade.nome as cidade, supermercado.nome as supermercado, produto.nome as produto, ofertas.valor, " +
            " ofertas.data_cadastro, supermercado.latitude, supermercado.longitude, " +
            " supermercado.codigo as cod_supermercado, ofertas.codigo as cod_oferta, " +
            " produto.codigo as cod_produto, rua.nome as rua, bairro.nome as bairro, " +
            " supermercado.numero as numero, cidade.estado as estado from ofertas " +
            " inner join produto on ofertas.cod_produto = produto.codigo " +
            " inner join supermercado on ofertas.cod_mercado = supermercado.codigo " +
            " inner join rua on supermercado.cod_rua = rua.codigo " +
            " inner join bairro on rua.cod_bairro = bairro.codigo " +
            " inner join cidade on bairro.cod_cidade = cidade.codigo " +
            " where "+data_hoje+" between data_inicial and data_final and produto.cod_barra = '" + cod_barra + "' order by valor";

        String dataCadastro,rua,estado,bairro,endereco;
        Integer numero;
        rs = stmt.executeQuery(sql);
        while (rs.next()) {

            OfertaModel oferta = new OfertaModel();

            oferta.setSupermercado(rs.getString("supermercado"));
            oferta.setProduto(rs.getString("produto"));
            oferta.setValor(rs.getBigDecimal("valor"));
            oferta.setCidadeSupermercado(rs.getString("cidade"));
            oferta.setData_cadastro(Validacoes.DataBancoParaPrograma(rs.getDate("data_cadastro").toString()));
            oferta.setLatitude(rs.getString("latitude"));
            oferta.setLongitude(rs.getString("longitude"));
            oferta.setCod_supermercado(rs.getInt("cod_supermercado"));
            oferta.setCod_oferta(rs.getInt("cod_oferta"));
            oferta.setCod_produto(rs.getInt("cod_produto"));
            oferta.setImg(localServidor+ rs.getString("img"));
            rua = rs.getString("rua");
            bairro = rs.getString("bairro");
            estado = rs.getString("estado");
            numero = rs.getInt("numero");
            endereco = "Rua : "+rua + " nº " + numero.toString() + ".\nBairro: " + bairro + ".\nCidade : " + rs.getString("cidade") + ", " + estado + ".";
            oferta.setEnderecoSupermercado(endereco);

            lista.add(oferta);
        }
    } catch (SQLException e) {
        System.out.println(e.getMessage());
        return null;
    }
    return lista;
}

```

Código Java na plataforma *Android* responsável por receber o conteúdo do WebService converte para Objetos e Retorna a lista de ofertas.

```
public List<OfertaModel> getListaOfertasCodBarra(String cod, String token) throws Exception {

    String metodo = "buscarofertascodbarra/";
    String[] resposta = new WebServiceCliente().get(URL_WS +metodo +cod+"/"+token);

    if (resposta[0].equals("200")) {
        Gson gson = new Gson();
        ArrayList<OfertaModel> listaCliente = new ArrayList<OfertaModel>();
        JsonParser parser = new JsonParser();
        JSONArray array = parser.parse(resposta[1]).getAsJSONArray();

        for (int i = 0; i < array.size(); i++) {
            listaCliente.add(gson.fromJson(array.get(i), OfertaModel.class));
        }
        return listaCliente;
    } else {
        throw new Exception(resposta[1]);
    }
}
```