

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA  
CELSO SUCKOW DA FONSECA – CEFET/RJ**

## **Fração Ideal – Um Sistema de Recomendação para o Mercado Imobiliário**

Angélica Cardozo de Araújo  
Rafael Cruz Stirling

Prof. Orientador: Rafael Castaneda Ribeiro

**Rio de Janeiro  
Agosto de 2011**

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA  
CELSO SUCKOW DA FONSECA – CEFET/RJ**

## **Fração Ideal – Um Sistema de Recomendação para o Mercado Imobiliário**

Angélica Cardozo de Araújo  
Rafael Cruz Stirling

Projeto final apresentado em cumprimento às  
normas do Departamento de Educação Superior  
do CEFET/RJ, como parte dos requisitos para obtenção  
do título de Tecnólogo em Sistemas para Internet

Prof. Orientador: Rafael Castaneda Ribeiro

**Rio de Janeiro  
Agosto de 2011**

A663 Araújo, Angélica Cardozo de  
Fração ideal – um sistema de recomendação para o mercado imobiliário  
/ Angélica Cardozo de Araújo [e] Rafael Cruz Stirling .—2011.  
x , 86f. : il. ; enc.

Projeto Final (Tecnólogo) Centro Federal de Educação Tecnológica  
Celso Suckow da Fonseca , 2011.  
Bibliografia : f.84 – 86  
Orientador : Rafael Castaneda Ribeiro

1.Engenharia de software 2.Sistemas de informação gerencial 3.Merca-  
do imobiliário I.Ribeiro, Rafael Castaneda (orient.)

CDD 005.1

## RESUMO

Recomendar bens ou informações para um usuário da WEB apresenta-se como um dos maiores desafios no mundo virtual. A recomendação adequada de um bem pode fazer a diferença entre conquistar o usuário ou perdê-lo. Devido a esta necessidade de conquista, a personalização tem se apresentado como um fator facilitador no momento de "cativar" o usuário. Apresenta uma visão geral sobre os sistemas de recomendação utilizados nas tarefas de personalização. Descreve as técnicas e estratégias de recomendação mais utilizadas, detalha a arquitetura dos sistemas e traz exemplos de utilização destes tanto no meio acadêmico quanto no mercado. Discute ainda o problema da privacidade e aborda tópicos que apontam para novas tendências na área.

Palavras-chave: Sistemas de Recomendação

## **ABSTRACT**

Recommend property or information for a Web user is presented as one of the biggest challenges in the virtual world. The recommendation of a suitable well can make the difference between win or you lose it. Due to this need for achievement, the customization has emerged as a facilitator at the time to "charm" the user. Presents an overview of the recommendation systems used in the customization tasks. Describes the most used techniques and strategies in recommendation, details the architecture of the systems and provides examples of use of both in academia and in the marketplace. It also discusses the problem of privacy and addresses topics that point to new trends in the area.

**Key-words:** Recommender Systems

## SUMÁRIO

1	Introdução .....	1
1.1	Motivação .....	1
1.2	Objetivo .....	2
1.3	Metodologia e Trabalhos Realizados .....	3
1.4	Organização do Trabalho .....	3
2	Fundamentação Teórica.....	4
2.1	Sistemas de Informação na Web e Comércio Eletrônico .....	4
2.2	Inteligência Artificial.....	6
2.3	Sistemas de Recomendação.....	8
2.3.1	Classificação de Sistemas de Recomendações .....	9
2.3.2	Filtragem Baseada em Conteúdo.....	10
2.3.2.1	Vantagens da Filtragem Baseada em Conteúdo .....	12
2.3.2.2	Desvantagens da Filtragem Baseada em Conteúdo .....	12
2.3.3	Filtragem Colaborativa.....	13
2.3.3.1	Vantagens da Filtragem Colaborativa .....	17
2.3.3.2	Desvantagens da Filtragem Colaborativa.....	17
2.3.4	Entrada/Saída.....	18
2.3.5	Métodos de Recomendações .....	19
2.3.6	Descobrimdo Hábitos de Consumo .....	20
2.3.7	Gerando Recomendações .....	21
3	Projeto do Software e Aspectos Técnicos .....	23
3.1	Contextualização .....	23
3.2	Metodologia de desenvolvimento.....	24
3.3	Especificação, Análise e Projeto do Sistema.....	24
3.3.1	Especificação dos requisitos.....	25
3.3.2	Especificação dos requisitos funcionais do sistema .....	25
3.4	Especificação dos demais requisitos do sistema .....	26
3.5	Modelo de Casos de Uso .....	26
3.5.1	Diagrama de Casos de Uso.....	26
3.5.2	Descrições textuais dos atores .....	27
3.6	Descrições textuais dos casos de uso.....	28
3.7	Modelo de Classes.....	36
3.7.1	Diagrama de Classes Conceituais.....	36
3.8	Glossário.....	37
3.9	Visões de Classes Participantes .....	38
3.10	Modelo de Interações .....	45
3.11	Descrição textual dos contratos das operações de sistema .....	45
3.12	Diagramas de seqüência de cada cenário relevante dos casos de uso .....	47

3.13	Projeto de Banco de Dados.....	58
3.14	Projeto Lógico de Banco de Dados .....	58
3.15	Projeto Físico de Banco de Dados.....	63
3.16	Projeto da Interface Gráfica.....	65
3.17	Hierarquia das telas e mapa de navegação .....	65
4	Aspectos de Implementação .....	68
4.1	Visão geral da arquitetura da aplicação.....	68
4.2	Estruturação da camada de apresentação .....	71
4.3	Estruturação da camada de negócio .....	73
4.4	Estruturação da camada de persistência .....	78
4.5	Outros Aspectos.....	80
4.5.1	Logging.....	80
4.5.2	Atributos Utilizados no Processo de Recomendação .....	81
4.5.3	Procedimentos de Implantação.....	82
5	Conclusão .....	83
5.1	Análise Retrospectiva.....	83
5.2	Trabalhos Futuros.....	83

## LISTA DE FIGURAS

Figura 1: Taxonomia de Sistemas de Recomendação no âmbito da Inteligência Artificial. ....	7
Figura 2: Classificação .....	10
Figura 3: Exemplo prático de aplicação do K-NN .....	16
Figura 4: Funcionamento de um Sistema de Recomendação .....	22
Figura 5: Diagrama de casos de uso .....	27
Figura 6: Diagrama de classes conceituais .....	37
Figura 7: VCP Manter Avulso .....	39
Figura 8: VCP Manter Avulso .....	40
Figura 9: VCP Manter Comercial .....	41
Figura 10: VCP Manter Edifício Comercial .....	42
Figura 11: VCP Manter Incorporador .....	43
Figura 12: VCP Manter Residencial .....	44
Figura 13: VCP Visualizar Imóveis Recomendados .....	45
Figura 14: Diagrama de Sequencia do caso de uso Manter (listar) Residencial. ....	47
Figura 15: Diagrama de Sequencia do caso de uso Manter (listar) Incorporador. ....	47
Figura 16: Diagrama de Sequencia do caso de uso Manter (listar) Avulso .....	48
Figura 17: Diagrama de Sequencia do caso de uso Manter (listar) Comercial. ....	48
Figura 18: Diagrama de Sequencia do caso de uso Manter Edifício (listar) Comercial. ....	49
Figura 19: Diagrama de Sequencia do caso de uso Manter (salvar) Avulso .....	50
Figura 20: Diagrama de Sequencia do caso de uso Manter (salvar) Comercial. ....	51
Figura 21: Diagrama de Sequencia do caso de uso Manter (salvar) Edifício Comercial. ....	52
Figura 22: Diagrama de Sequencia do caso de uso Manter (salvar) Residencial. ....	53
Figura 23: Diagrama de Sequencia do caso de uso Manter (salvar) Incorporador .....	54
Figura 24: Diagrama de Sequencia do caso de uso Manter Cesta de Imóveis .....	55
Figura 25: Diagrama de Sequencia do caso de uso Buscar Imóveis .....	56
Figura 26: Diagrama de Sequencia do caso de uso Buscar Imóveis .....	57
Figura 27: Modelo entidade-relacionamento .....	64
Figura 28: Hierarquia das telas na perspectiva do cliente. ....	65
Figura 29: Hierarquia das telas na perspectiva do corretor. ....	65
Figura 30: Mapa de navegação para o caso de uso Manter Cesta de Imóveis .....	66
Figura 31: Mapa de navegação para o caso de uso Manter Avulsos .....	67
Figura 32: Mapa de navegação para o caso de uso Manter Incorporador. ....	67
Figura 33: Retirada do site do framework Yii .....	69
Figura 34: Processamento de requisições .....	70
Figura 35: Menu da Aplicação. ....	72
Figura 36: Automação dos componentes .....	72
Figura 37: Formulário de informações de um imóvel avulso .....	73
Figura 38: Interface de busca de imóveis. ....	73
Figura 39: Classe Residencial .....	75
Figura 40: Classe Carrinho .....	76



Figura 41: Classe de Monitoração de Busca.....	77
Figura 42: Trecho de código com o mapeamento para o banco de dados MySql.....	79
Figura 43: Código de mapeamento entre uma classe Active Record e sua respectiva tabela. .	79
Figura 44: Trecho de código com a configuração do Log.....	80

## **LISTA DE ABREVIATURAS E SIGLAS**

WEB – World Wide Web

FC – Filtragem Colaborativa

IA – Inteligência Artificial

FBC – Filtragem Baseada em Conteúdo

URL - Uniform Resource Locator

MVC – Model-View-Controller

FTP – File Transfer Protocol (protocolo de transferência de arquivos)

SGBD – Sistema Gerenciador de Banco de Dados

# Capítulo 1

## Introdução

O avanço das tecnologias atuais transformou a *WEB* em um grande conjunto de ferramentas, que é cada vez mais utilizado para que as pessoas busquem melhorias em sua qualidade de vida. Estas ferramentas estão ao alcance da maioria em *sites* que na maioria das vezes não evoluem de acordo com a necessidade de seus usuários. Um usuário comum pode expressar suas necessidades sobre um determinado produto ou serviço, porém dependendo da situação despertar interesse por outro de melhor qualidade e/ou valor, enfim, atendendo melhor sua necessidade. As variações constantes de mercado cada vez mais exigem das organizações um nível elevado de qualidade e produtividade, sem que haja impacto nos custos para tal.

Uma situação bastante comum é quando desejamos um bem por uma quantia monetária determinada, buscamos por este bem e o encontramos. Como nos últimos anos, deu-se um crescimento explosivo no volume de informações online disponíveis na Internet [1], este tipo de busca é bastante dispendiosa para a capacidade humana dada a imensa gama de resultados que não possuem relevância com o foco da consulta.

Neste momento, a aplicação de metodologias de filtragem de informações pode ser utilizada para alavancar negócios de maior porte alcançando satisfação para ambas as partes, os Sistemas de Recomendação apresentam-se como uma proposta desse tipo de tecnologia, que pode auxiliar durante estas consultas destacando somente as informações relevantes e ocultando as que não interessam.

### 1.1 Motivação

A proposta dos Sistemas de Recomendação é tentar prever as opções de itens (como livros, músicas, filmes, notícias e páginas de Internet) nos quais o usuário possa estar interessado, baseando-se em informações provenientes principalmente do seu perfil [1].

Os Sistemas de Recomendação são geralmente utilizados para diminuir o impacto causado pelo problema da sobrecarga de informação. Seu objetivo é apontar itens que possam despertar interesse ao usuário, para assim realizar a recomendação [2]. Como exemplos desta aplicação temos a recomendação de produtos na Amazon.com [3], livros no sistema LIBRA [4] e de filmes no MovieLens.org [5]

Estes Sistemas de Recomendação são utilizados no comércio eletrônico para personalizar serviços trazendo benefícios tanto para empresas e consumidores. Algumas empresas como Amazon.com, Netflix.com, e Procter & Gamble lançam mão do artifício das tecnologias de recomendação, com o objetivo de incrementarem seus negócios na Internet [6]. KONSTAN e RIEDL [6] durante uma análise na intenção de quantificar o benefício gerado na utilização de um Sistema de Recomendação obtiveram como resultado após agregar um Sistema no *site* GUS - *www.gus.co.uk*, o incremento de 60% na venda cruzada de produtos.

Para o consumidor, os Serviços de Recomendação prestam auxílio e reduzem esforços necessários na identificação dos produtos de interesse em grandes *sites* da *WEB* de comércio eletrônico, como por exemplo, no *site* Amazon.com que dispõe de mais de 2 milhões de itens disponibilizados. Ao adicionar itens semelhantes ao resultado da busca é possível gerar uma opção, aumentando assim a probabilidade de um negócio de maior volume.

## 1.2 Objetivo

O objetivo deste trabalho é agregar informações relacionadas que possam ampliar as opções durante a decisão de compra através do uso de um Sistema de Recomendação que aplica uma técnica de filtragem baseada em conteúdo. Esse processo foi idealizado com o intuito de auxiliar usuários que buscam imóveis exibindo ofertas de imóveis através de recomendações de imóveis com características semelhantes.

Como protótipo de aplicação apresenta-se um sistema que auxilia potenciais compradores de imóveis na pesquisa de ofertas. Como objetivos específicos, este trabalho apresenta o desenvolvimento de agentes de software para extração de conteúdo,

recomendação de itens e descoberta de tendências para atender demandas por imóveis em regiões específicas.

### **1.3 Metodologia e Trabalhos Realizados**

Analisar e integrar o conteúdo selecionado pelo usuário durante sua navegação para utilizá-lo nas recomendações dos imóveis, auxiliando usuários que possam apresentar incertezas durante o processo de negociação da compra de imóveis é a maior contribuição deste trabalho. O domínio do sistema desenvolvido abrange o mercado imobiliário, porém é possível adaptá-lo e aplicá-lo a outros domínios e aplicações.

### **1.4 Organização do Trabalho**

A estrutura do trabalho está definida em quatro capítulos, sendo o primeiro a Introdução e o último a Conclusão. No capítulo 2 apresentam-se os conceitos relativos a Sistemas de Recomendação. No capítulo 3 será apresentado o projeto do software e aspectos técnicos envolvidos no desenvolvimento do sistema, sendo que a retrospectiva sobre o processo de desenvolvimento do sistema e trabalhos futuros será apresentada no capítulo 4.

## Capítulo 2

### Fundamentação Teórica

Nas próximas seções serão apresentados os conceitos de Sistemas de Informação, Comércio Eletrônico e Sistemas de Recomendação, que moldam a base conceitual para a construção do sistema. Como o sistema proposto irá utilizar técnicas de Inteligência Artificial (IA), mais especificamente de Sistemas de Recomendação, alguns conceitos pertinentes a esses assuntos serão apresentados na seção 2.2.

#### 2.1 Sistemas de Informação na Web e Comércio Eletrônico

Com os recursos oferecidos pelos Sistemas de Informação na Web, é possível obter acesso à informação e propiciar mudanças de comportamento de seus usuários. Uma forma de obter acesso à informação e auxiliar na mudança de comportamento dos usuários pode ser obtida com os recursos oferecidos pela Web simplesmente fazendo uso de Sistemas de Informação. Uma ferramenta onde seus processos são executados por meio de um navegador (*browser*) e que permite seu acesso via Internet, define um Sistema de Informação na Web [7].

Os sistemas baseados em Web apresentam uma quantidade significativa de vantagens em relação a sistemas baseados em *software desktop*, que por sua vez precisam residir no computador do usuário, ocupando espaço de armazenamento e muitas vezes, exigindo um hardware de alto desempenho para sua execução. Como vantagem, tem-se a manutenção e atualização que são feitas de forma transparente, sem causar impactos ao usuário, bastando ao administrador do sistema publicar a nova versão no servidor de aplicações WEB e imediatamente todos os usuários têm acesso ao conteúdo atualizado e funcionalidades novas. A necessidade de se ter uma conexão com a Internet sempre que for preciso acessar a aplicação pode ser citada como uma desvantagem. A Internet fomentou o surgimento e desenvolvimento do comércio eletrônico. Também definem [8] comércio eletrônico como um processo de compra, venda e troca de produtos, serviços e informações por redes de computadores ou pela própria Internet.

Conforme definido por [9] hoje em dia, todos os negócios competem em dois mundos: um mundo físico, de recursos e gerentes que podem ser vistos e manipulados e, um mundo virtual, feito de informação. O segundo tem possibilitado o crescimento do comércio eletrônico, ao qual os autores se referem como *marketspace*, para distingui-lo do mundo físico, que é denominado *marketplace*. Eles definem *marketspace* como sendo uma área virtual onde produtos e serviços existem como informação digital e podem ser comprados através de canais baseados em informação.

Encontram-se no comércio eletrônico ofertas de diversos produtos. Nas emissoras de televisão, outdoors, rádios, jomais e outros meios de comunicação, utiliza-se o marketing em massa, para atrair a atenção de uma grande quantidade de pessoas. Neste formato, a abrangência é enorme, porém sua eficiência é baixa. Por outro lado, o marketing no comércio eletrônico é um marketing direto, utilizando-se a personalização de ofertas de produtos para seus usuários, atraindo desta forma novos usuários e ao mesmo tempo tomando-os fiéis.

O consumidor brasileiro vem ganhando confiança em comprar pela Internet. Recente pesquisa divulgada pelo Banco Itaú, a E-bit, empresa que pesquisa o comércio virtual, e a Câmara Brasileira de Comércio eletrônico (câmara - e.net) mostrou um crescimento de 76% no volume de transações por meio do e-commerce em 2006, em relação ao ano anterior que foram de R\$4,4 bilhões [10].

Preços competitivos, promoções, descontos e brindes são fatores determinantes para motivar a aquisição de mais produtos pela Internet, para um público mais jovem (até 35 anos) de acordo com a pesquisa. Estes são quase 70% dos motivos que fazem com que realizem mais ou menos compras no canal. Já o público de idade mais avançada (entre 36 e 59 anos) tem preocupação com a segurança na hora da transação. Este é o fator que determina o número de compras a serem feitas.

A pesquisa revela ainda que 50% dos usuários até 35 anos apresentam-se mais ansiosos e exigentes com relação ao prazo de entrega dos produtos adquiridos no canal do que o público mais velho, principalmente por efetuar o pagamento antes de receber a mercadoria. Depois de ouvir cerca de 700 lojistas virtuais, a E-bit concluiu em sua pesquisa que as expectativas dos empresários do setor para o ano de 2007 são de um crescimento de 2,8

milhões no número de pessoas comprando pela Internet. Em 2006 foram 7 milhões de brasileiros que fizeram pelo menos uma compra pela Internet.

O Brasil ocupa o décimo lugar no ranking mundial em número de usuários de Internet, mas percentualmente perde para a Argentina, onde 10 milhões acessam a rede (26,4% de penetração), Estados Unidos, que registram 204 milhões de internetautas (68,1% de todo o seu contingente populacional) segundo a E-bit. Estas informações demonstram a importância do comércio eletrônico na economia e as grandes possibilidades que esta área de desenvolvimento apresenta para os profissionais da Tecnologia da Informação (TI). Além disso, demonstram a relevância que o sistema proposto pode alcançar para os proprietários de lojas virtuais de Informática e seus consumidores.

## **2.2 Inteligência Artificial**

O que diferencia a inteligência artificial das demais ciências que estudam o comportamento humano é que a primeira não objetiva apenas compreender os fatores que influenciam o ser humano a tomar decisões ou agir racionalmente, mas também construir modelos inteligentes a partir destas informações.

“A Inteligência Artificial sistematiza e automatiza tarefas intelectuais e, portanto é potencialmente relevante para qualquer esfera da atividade intelectual humana. Neste sentido, ela é verdadeiramente um campo universal” [11].

Na busca pela simplificação de problemas complexos ligados ao comportamento humano a inteligência artificial possibilita a abordagem de quatro métodos para se obter informações que caracterizem o comportamento de uma pessoa e depois reproduzi-las em entidades artificiais. Estas abordagens se baseiam em agir de forma humana, pensar de forma humana, pensar racionalmente e agir racionalmente [11].

É de extrema dificuldade elaborar uma definição de Inteligência Artificial, a partir do instante em que se verifica que a própria inteligência humana, não artificial, encontra já um vasto campo de conceituações, nem todas elas convergentes [12].

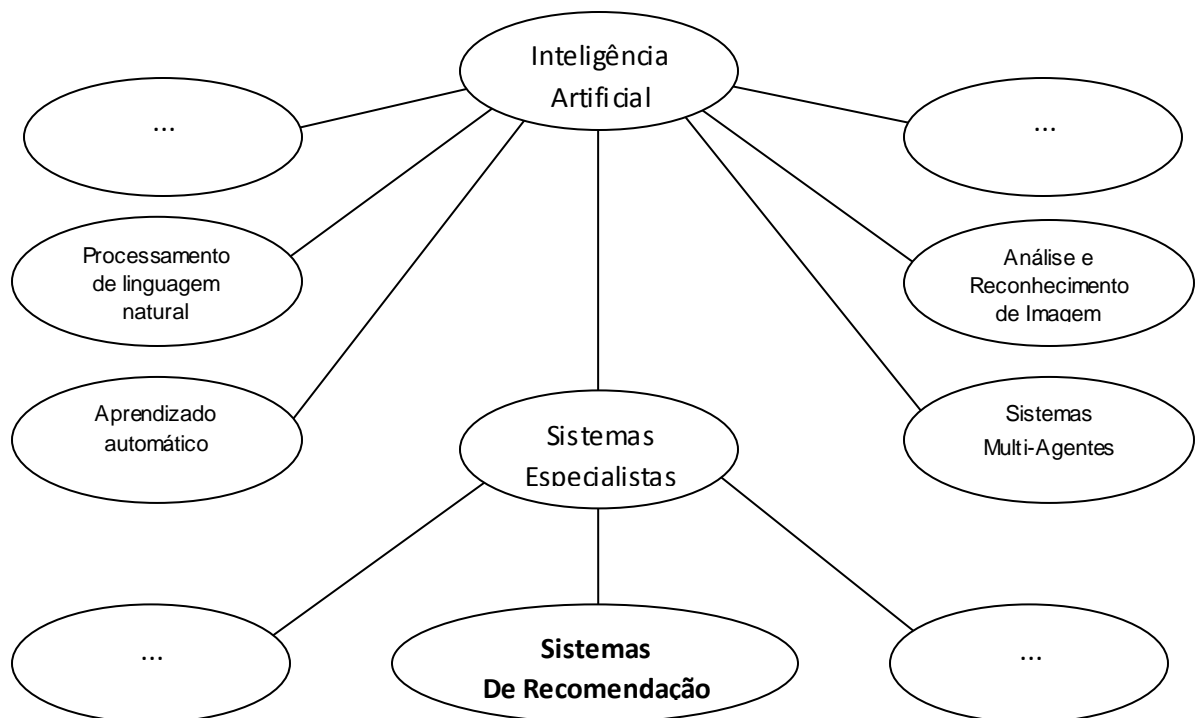


ROSSINI (1988 apud LEVINE, 2003) define Inteligência Artificial como:

"(...) simplesmente uma maneira de fazer o computador pensar inteligentemente. Isso é conseguido estudando como as pessoas pensam quando estão tentando tomar decisões e resolver problemas, dividindo esses processos de pensamento em etapas básicas e desenhando um programa de computador que solucione problemas usando essas etapas. Inteligência Artificial então fornece um método simples e estruturado de se projetar programas complexos de tomada de decisão".

A Inteligência Artificial pode ser compreendida como o conjunto de várias técnicas e metodologias de programação usadas para tentar resolver os problemas de forma mais eficiente que soluções algorítmicas, e fazendo isso o mais próximo possível de um ser humano. A Figura 1 apresenta uma taxonomia com o objetivo de ilustrar algumas áreas onde é possível aplicar técnicas de Inteligência Artificial. Outro conceito de IA encontrado na literatura:

"Inteligência Artificial é o estudo de como fazer os computadores realizarem coisas que, no momento, as pessoas fazem melhor." [13].



**Figura 1:** Taxonomia de Sistemas de Recomendação no âmbito da Inteligência Artificial.

A seguir apresenta-se com mais detalhes o estudo da técnica de Sistemas de Recomendação, dentre as diversas técnicas de Inteligência Artificial esta se mostra mais adequada para o sistema proposto.

## **2.3 Sistemas de Recomendação**

A gama de informações produzida e disponibilizada através da Internet ocasionalmente pode gerar uma sobrecarga cognitiva sobre o usuário final. Devido a este problema várias tecnologias têm surgido para apoiar a seleção, recuperação e filtragem da informação desejada ou de interesse do usuário. Esta atividade é na maioria das vezes, realizada por uma apresentação explícita das necessidades do usuário, ou seja, através do uso de palavras-chave.

Tendo como motivação esta necessidade de tecnologias que auxiliem as pessoas a percorrerem um grande volume de informações, ocultando o que é irrelevante, e exibindo apenas o que é relevante, originam-se os Sistemas de Recomendação [2].

A linha de pensamento seguida por [1] apresenta como definição dos Sistemas de Recomendação, sistemas que utilizam as opiniões de uma comunidade de usuários a fim de auxiliar indivíduos que participam desta mesma comunidade a identificar conteúdos de interesse relevante com base em um conjunto de alternativas, evitando assim a sobrecarga de informações.

Os Sistemas de Recomendação tem a função de auxiliar na eficácia do processo de indicação da mesma maneira como ocorre na no relacionamento social entre seres humanos [1]. Seu funcionamento típico pode ser descrito como sistema onde as pessoas (usuários) fornecem recomendações como entradas que são agregadas e direciona para os indivíduos considerados potenciais interessados neste tipo de recomendação. Definir e descobrir este relacionamento de interesses é um dos grandes desafios deste tipo de sistema, que consiste em realizar o alinhamento correto entre os que estão recomendando e aqueles que estão recebendo a recomendação [2].

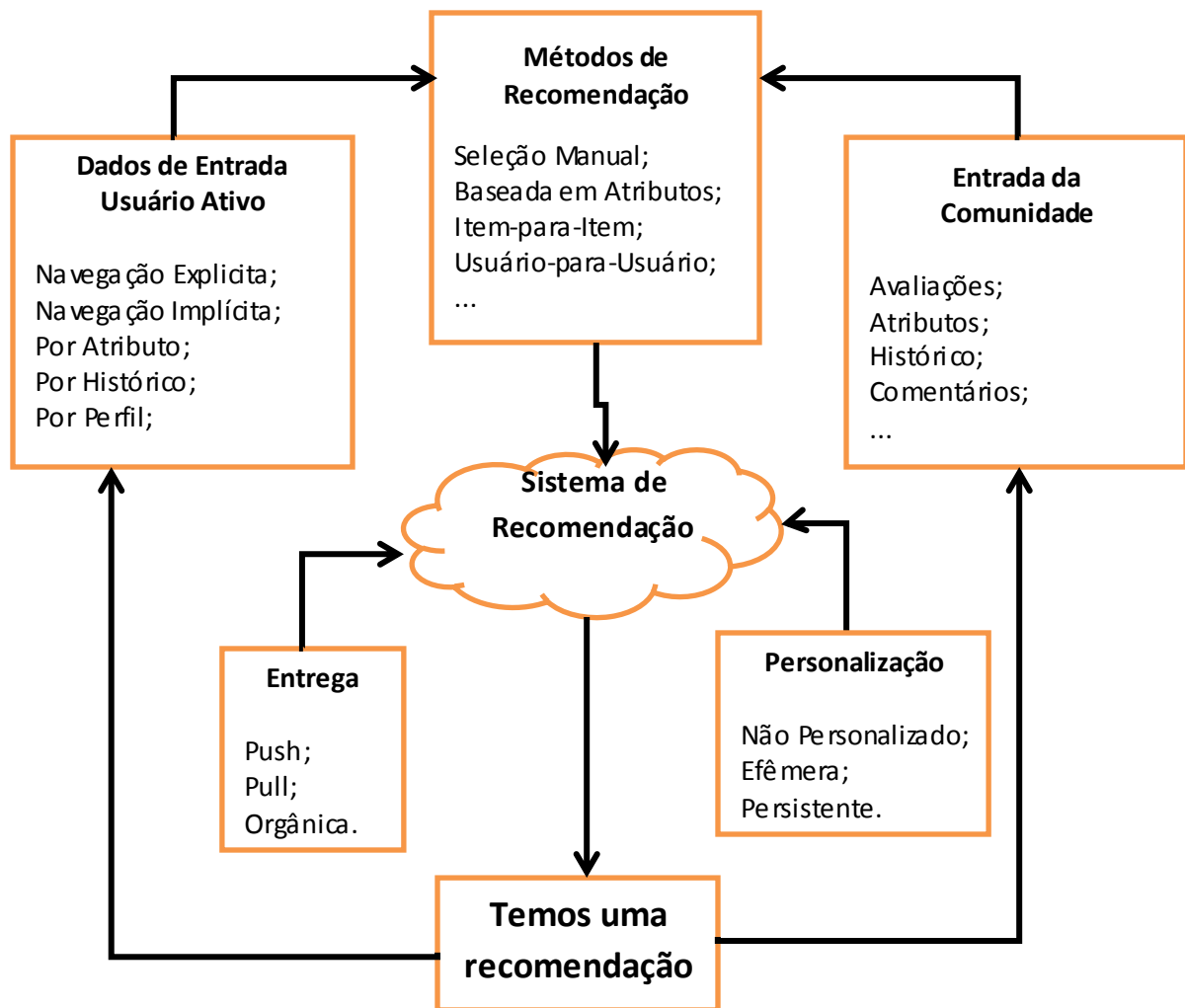
CAZELLA (2001 apud SCHAFER, 2006) apresenta uma abordagem mais prática expondo que os Sistemas de Recomendação são usados pelos sites de comércio eletrônico para sugerir produtos para seus usuários e fornecer informações que procuram auxiliar os usuários sobre qual produto deve ser adquirido.

O uso de Sistemas de Recomendação não é novidade, são amplamente utilizados em aplicações de empresas e na Internet. Normalmente estes sistemas lidam com aplicações que unem as entidades denominadas usuários e itens. O ambiente ideal para realizar a personalização é a Internet. Na Internet, as páginas não possuem os problemas da mídia impressa, como o layout de propagandas e custo de impressão. Este meio também possibilita o rastreamento dos hábitos de navegação e consumo de seus usuários, o que é impossível em outros meios de comunicação [2].

As páginas são geradas dinamicamente por um servidor WEB por meio da interação do usuário com seu navegador da Internet. Durante o processo, o servidor acessa o Sistema de Recomendação com a identificação do usuário, que retorna seu perfil e com base nesta informação gera as recomendações. As recomendações são enviadas ao servidor WEB para criar uma página dinâmica contendo as informações de acordo com o perfil do usuário. Esse processo é contínuo e sempre ocorrerá quando o usuário acessar uma página que exiba conteúdo personalizado [7].

### **2.3.1 Classificação de Sistemas de Recomendações**

SCHAFER et al. [14] idealizou uma taxonomia para classificar a arquitetura de um sistema de recomendação. Este modelo proposto por SCHAFER foi ampliado [7] e alguns pontos novos foram acrescentados. A arquitetura divide-se em três blocos, onde cada um possui sua complexidade de implementações e classificações. O intuito desta divisão é aumentar o entendimento desses sistemas uma vez que cada bloco é um componente importante do sistema de recomendação, conforme demonstrado pela Figura 2.



**Figura 2:** Classificação [7]

### 2.3.2 Filtragem Baseada em Conteúdo

Por bastante tempo os cientistas têm direcionado esforços para minorar as dificuldades criadas pela sobrecarga de informações tendo como base tecnologias que reconhecem e categorizam informações, conforme demonstrado por HERLOCKER [15]. Os sistemas baseados nesta técnica têm como objetivo gerar de forma automática, descrições dos conteúdos dos itens e comparar esta descrição de cada item com a descrição dos interesses dos usuários ou com o histórico de consumo do mesmo, no intuito de verificar se o item é interessante para o usuário [15].

Através de informações fornecidas pelo próprio usuário, através do perfil do usuário, através de uma consulta, ou aprendendo com os itens que o usuário consome (aqueles que o usuário se visualiza e adquire) é possível obter descrição de interesses do usuário. Este tipo de técnica que realiza filtragem baseada na análise de conteúdo do item e no perfil do usuário foi denominada de filtragem baseada em conteúdo (FBC) [15].

Ao contrário da filtragem colaborativa, a filtragem baseada em conteúdo faz as comparações de similaridade entre o usuário e um item vendável do sistema. Assim, os itens que mais se assemelhem ao perfil do usuário. A filtragem colaborativa parte do princípio que se o usuário desperta interesse por um produto, gostará de produtos semelhantes. Os produtos que podem ser classificados utilizando filtragem baseada em conteúdo devem possuir informações textuais suficientes sobre suas características [7].

Este método tem como um de seus pontos principais o perfil do usuário, onde sua criação é de total importância e indispensável. Através deste perfil é possível recuperar as características textuais (palavras-chave) dos produtos que o usuário tem, ou não, interesse. Os textos compõem a descrição dos produtos e assim definem seu conteúdo.

Um exemplo de algoritmo bastante utilizado atualmente em implementações de Filtragem Baseada em Conteúdo é técnica TF-IDF (*Term-frequency Inverse-Document-Frequency*). Esta técnica calcula a similaridade de textos baseada na frequência com que as palavras-chave são encontradas em um conjunto ou corpo de documentos. O uso desta técnica pode ser aplicado na recomendação de textos, como exemplo, em textos de informática em que as palavras software e Java possuem maior frequência e destacam-se das demais [7]. Os textos com maior incidência das palavras-chave possuem maior relevância e podem ser utilizados como conjunto de recomendações.

Outra técnica que também tem sido bastante utilizada atualmente é a técnica que forma o perfil do usuário baseada em análise simbólica das variáveis que descrevem o produto. O perfil do usuário é formulado com variáveis multivaloradas, onde uma palavra-chave suporte é associada a um peso que representa o peso dessa variável no perfil do usuário. Nessa técnica é levada em conta a nota que o usuário forneceu para o produto, sendo construído o perfil positivo, para produtos que o agradaram o usuário, e o negativo, para produtos que o usuário não gostou [7].

A filtragem baseada em conteúdo não ocorre a formação de vizinhança entre usuários que possuem as mesmas preferências. A identificação dos produtos mais similares ao perfil deste usuário é obtida através de comparações entre o perfil do usuário e todos os produtos disponíveis para a venda. O perfil do usuário deve conter os textos que descrevem os produtos que o despertam interesse. O que se leva em consideração é a frequência com que as palavras-chave estão contidas nas descrições dos produtos. Esta técnica também discrimina as palavras que podem ou não descrever certo tipo de texto. A palavra tecnologia nada diz do seu contexto, mas se for a palavra "JAVA", por exemplo, é possível ver que se trata de um texto sobre da tecnologia da informação.

As comparações são feitas para cada tipo de atributo que compõe os perfis do usuário e do produto. Comparando cada tipo de atributo com o mesmo tipo de atributo no outro perfil. Por exemplo, se um atributo descreve a categoria de um filme, como drama, esse atributo vai ser comparado com o atributo que também descreve a categoria do filme.

### **2.3.2.1 Vantagens da Filtragem Baseada em Conteúdo**

De acordo com TORRES [7], temos como as principais vantagens da filtragem baseada em conteúdo:

- Ausência do problema do primeiro avaliador: Por não precisar que algum usuário avalie um produto para gerar a recomendação, a filtragem baseada em conteúdo não possui o problema do primeiro avaliador;
- A capacidade de recomendar todos os itens: Os produtos disponíveis no banco de dados são todos comparados com o perfil do usuário, assim, todos os produtos possuem chances de serem recomendados ao usuário;
- O usuário recebe recomendação do que deseja conforme seu perfil formado no sistema: Outra vantagem é a redução da esparsidade, pois para todo texto e/ou atributo pode ser medida a similaridade com o perfil do usuário.

### **2.3.2.2 Desvantagens da Filtragem Baseada em Conteúdo**

As principais desvantagens na filtragem baseada em conteúdo são [7]:

- Impossibilidade de análise de conteúdo em domínios não textuais: Apesar de a técnica analisar o conteúdo de um texto, sua qualidade, clareza ou elegância não são levadas em conta em sistemas que utilizam essa técnica, e como o usuário não fornece notas nessa filtragem, fica mais difícil ainda para um sistema saber se o usuário achou esse um texto bem escrito ou não, além disso, não é levado em conta o renome de quem produziu o texto;
- Não consideração de aspectos como qualidade do texto e renome do autor;
- Superespecialização: a superespecialização acontece porque os sistemas de recomendação de FBC fazem comparação de palavras-chave, então as recomendações geradas são semelhantes com as palavras-chave encontradas no perfil do usuário.

A filtragem baseada em conteúdo também possui o problema de ser completamente baseada em palavras, ou seja, não existe possibilidade de tentativa do sistema para tentar interpretar e recomendar imagens, músicas, vídeos a fim de sugerir ao usuário. Em casos que as palavras sejam de descrição de atributos e/ou características técnicas de algum produto, não impede um sistema de filtragem baseada em conteúdo de funcionar, pois são palavras. Em textos que possuem as palavras "carro" e "automóvel" podem não ser considerados similares, mesmo que abordem o mesmo assunto. A solução é utilizar dicionários de sinônimos para identificar quais palavras têm o mesmo significado ou não utilizar palavras com sentidos iguais mais com escrita diferentes.

### **2.3.3 Filtragem Colaborativa**

A filtragem colaborativa não se constitui em algo novo e apresenta uma proposta diferenciada dos sistemas de filtragem uma vez que não requerem compreensão ou conhecimento de conteúdo dos itens [16] e foi proposta para atender as situações que estavam em aberto na filtragem baseada em conteúdo.

A filtragem colaborativa parte do princípio de que pessoas buscam por recomendações sobre produtos que irão consumir que por sua vez são criadas por outras pessoas que tenham

gostos parecidos com elas. Nos sistemas colaborativos a essência está na troca de experiências entre pessoas que possuem interesses comuns. Nestes sistemas os itens são filtrados baseando-se na avaliação feita pelos usuários daqueles itens, ao invés do conteúdo dos itens [16].

Um dos primeiros sistemas de recomendação a utilizar a técnica de filtragem colaborativa foi sistema Tapestry [1], que recomendava e-mails recebidos de diversas listas e utilizavam filtragem colaborativa para organizar estes e-mails de acordo com o interesse do usuário. Após esse trabalho foi desenvolvido o sistema GroupLens [17] que impulsionou a pesquisa dessa área sendo citado por diversos outros trabalhos. Esse sistema recomendava notícias avaliadas quantitativamente por notas de 1 a 5 pelos usuários, relacionando o quanto o usuário gostou daquela notícia e utilizando a técnica de filtragem colaborativa para apresentar notícias personalizadas.

Identificar indivíduos que possuem preferências comuns é a motivação da filtragem colaborativa. Apontando esses indivíduos, forma-se um grupo de indivíduos com gostos comuns, depois os indivíduos desse grupo colaboram entre si através dos produtos que os mesmos gostam para encontrar um produto que um usuário-alvo ainda não possua. O produto, ou os produtos são então recomendados a este usuário, que provavelmente irá julgar interessante o modo como a sugestão dos produtos foi realizada [7].

Existem 3 etapas para descrever a técnica de filtragem colaborativa [16]:

- Métrica de similaridade: Consiste em calcular o peso de cada usuário em relação à similaridade ao usuário ativo;
- Selecionar um subconjunto de usuários com maiores similaridades (denominados: vizinhos) para considerar na predição;
- Normalizar as avaliações e computar as predições ponderando as avaliações dos vizinhos com seus pesos.

Uma possível implementação desta técnica é chamada de "*k-nearest-neighbor*" (*K-NN*) que fará parte da abordagem do sistema. No primeiro passo, a definição da similaridade pode ser realizada através de diversas técnicas, como por exemplo, o cálculo da distância Euclidiana, função vista na Equação (1) [18]. A identificação da vizinhança é medida a



similaridade de um item de com outros itens de um determinado conjunto. A equação (1) calcula a distância entre dois vetores que representa o item determinado. Desta maneira quanto menor a distância calculada de um vetor a outro vetor, mais semelhantes são os itens. O valor da distância varia de 0 (mais similar) a 1(menos similar).

$$d(X,Y)=\sqrt{\sum_{i=1}^n (X_i-Y_i)^2}$$
(1)

Uma forma de exemplificar o funcionamento do algoritmo KNN, pode ser descrita como:

```

Input: Quantidade de Vizinhos (K)
Input: Registro Selecionado (RS)
Input: Registros Candidatos (RCS)
Output: Registros mais similares (RMS)
If Há RS do
  Cria um Vetor de Distâncias (VD);
  while Há registros em RCS do
    Registro Candidato (RC) = Retira de RCS;
    Distancia (D) = Calcula a Distância entre RS e RC;
    Insere D em VD ordenado;
  end
  RMS = Seleção dos K doadores com menor distância;
  Retorna RMS;
end

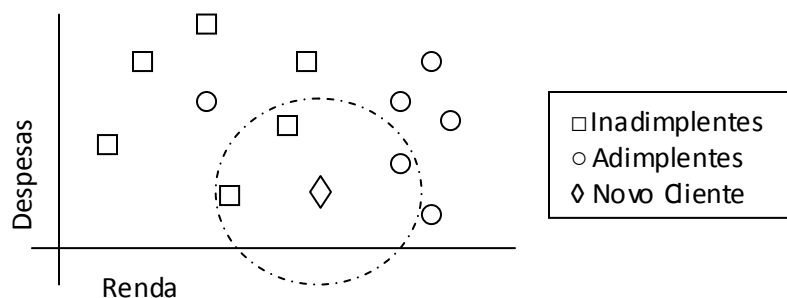
```

Uma exemplificação dos passos compreendidos no processo de classificação do método *K-NN* pode ser descrito como:

- Calcular a distância do novo registro para cada um dos registros contidos no grupo selecionado como referência utilizando uma métrica de distância, como por exemplo, a distância euclidiana;

- Apontar os  $k$  registros contidos no grupo de referência com menor distância tendo o novo registro como referência;
- Identificar a classe mais freqüente dentre o grupo de  $k$  registros apontados na etapa anterior;
- Comparar a classe identificada com a classe real do registro, computando erro ou acerto do algoritmo (passo necessário apenas para avaliação do desempenho do método  $K$ -NN na base de referência, normalmente o algoritmo é encerrado no passo anterior).

Uma forma de ilustrar graficamente o funcionamento pode ser apresentada como o exemplo do contexto de análise de crédito. A base de dados é representada pela Figura 3. O conjunto foi dividido em duas classes. A classe de clientes inadimplentes foi representada por um quadrado e a classe de clientes adimplentes com um círculo. O objetivo é avaliar a possibilidade de concessão de crédito para um novo cliente.



**Figura 3:** Exemplo prático de aplicação do K-NN [18]

Conforme o exemplo foi calculado o valor das distâncias entre o novo registro e todos os registros existentes. Assumindo que o número de vizinhos mais próximos seja 3, somente os 3 registros mais próximos foram selecionados. Foi possível observar que a classe de maior incidência dentro da área demarcada pelo algoritmo foi a classe de clientes inadimplentes, portanto a concessão de crédito ao novo cliente possui um alto risco de investimento e seu crédito seria negado.

### 2.3.3.1 Vantagens da Filtragem Colaborativa

O uso da Filtragem Colaborativa possui as seguintes vantagens:

- Independência de conteúdo;
- Geração de recomendações baseadas em preferências dos usuários;
- Possibilidade de produzir recomendações inesperadas e de alta qualidade.

O embasamento da técnica de Filtragem Colaborativa é fundamentado nas avaliações de usuários para gerar a recomendação. Essas avaliações fornecem ao sistema uma independência de conteúdo, gerando recomendações de diversos tipos de produtos, notícias, músicas, artigos científicos, entre outros. A capacidade de produzir recomendações de qualidade depende das avaliações positivas e negativas fornecidas por outros usuários. Para um sistema automatizado é extremamente complexa tarefa de julgar a qualidade de um livro, porém quando há interação de um ou mais usuários avaliando esse livro a tarefa torna-se mais simples, permitindo ao sistema produzir recomendações de qualidade e recomendar com mais precisão para outros usuários. A recomendação chamada de *serendipity*, ou seja, capacidade de o sistema recomendar produtos inesperados ao usuário, uma recomendação de grande qualidade compreende a terceira vantagem.

### 2.3.3.2 Desvantagens da Filtragem Colaborativa

São as principais desvantagens da técnica de Filtragem Colaborativa:

- Dispersão da base de dados: Os sites de comércio eletrônico possuem uma base de dados, geralmente grande. Para formar um perfil é preciso que o usuário compre ou avalie muitos produtos. Isso gera um problema chamado de dispersão dos dados, assim, é difícil encontrar bons vizinhos pelo baixo número de produtos em comum, esta situação se chama de baixo *overlapping*;
- Problema do 1º avaliador: A base da FC é a avaliação de produtos por usuários, então, um produto que não tenha sido avaliado, nunca será recomendado, assim, esse produto nunca irá aparecer como opção de recomendação para o sistema de recomendação para os demais usuários;

- Ovelha negra: O problema do falso vizinho ou ovelha negra acontece quando usuários são reconhecidos como semelhantes para o sistema, mas na realidade, esses usuários não possuem preferências parecidas para outros itens. Pode ocorrer que usuários, avaliem alguns itens iguais e nos perfis dos mesmos possuam poucos itens. Usuários com mais produtos avaliados igualmente têm muito mais chance de realmente serem semelhantes. O que se pode fazer é multiplicar por uma constante  $n/50$  quando o número de itens avaliados for inferior a 50, onde  $n$  é número de produtos avaliados. Usuários com mais de 50 itens avaliados não são influenciados na sua similaridade [17].

### 2.3.4 Entrada/Saída

Esta seção apresenta como se processa o fluxo das informações de entrada e saída de um sistema de recomendação. A classificação leva em conta três tipos de informações [7]:

- **Informações do usuário-alvo (ativo)**

Um sistema de recomendação necessita de informações do usuário alvo por se tratar de uma personalização para o próprio usuário; quanto mais informações forem obtidas sobre esse usuário, mais o sistema poderá recomendar produtos para ele de acordo com suas preferências.

Atualmente é comum o cadastro de usuários em *sites* de comércio eletrônico, fóruns, entre outros. Pode-se utilizar *cookies*, que armazenam informações sobre o usuário ligado àquele computador.

- **Informações da comunidade de usuários**

As informações da comunidade são coletadas de todos os usuários que utilizam o *site*, através da quantidade de produtos vendidos ou os mais visualizados. Podem ser utilizadas para calcular recomendações de produtos aos usuários. Apesar de não serem consideradas recomendações personalizadas, são usadas por diversas lojas em seus *sites*.

- **Informações de saída**

A saída representa a resposta do sistema, variando em tipo, quantidade e aparência. A saída mais comum é a sugestão, que aparece na forma: “Temos uma recomendação para você!” ou ainda “Experimente este produto!”.

## 2.3.5 Métodos de Recomendações

Os métodos que são utilizados atualmente para gerar recomendações representam as diferentes maneiras que as informações coletadas no perfil do usuário são utilizadas para gerar recomendações, destacando-se [7]:

- **Seleção manual**

A seleção manual é o método de recomendação de menor complexidade, onde especialistas propõem as recomendações com base em um conhecimento prévio do público-alvo.

- **Resumos estatísticos**

Os resumos estatísticos constituem um método de recomendação que visa complementar os métodos de seleção manual, pois nem sempre editores e especialistas acertam as necessidades e preferências de todos os usuários. Exemplos de resumos estatísticos são índices de satisfação do usuário e índices de produtos mais vendidos.

- **Recuperação bruta**

Sistemas de recuperação bruta implementam uma interface simples para busca por palavras-chave numa base de dados. O usuário procura um produto ou serviço por uma palavra-chave e o *site* exibe uma lista dessa busca em sua base de dados, de acordo com essa palavra-chave. Atualmente esse método é bastante utilizado em *sites* de comércio eletrônico e não representa propriamente uma recomendação, já que esse sistema apenas recupera informações de uma base de dados e não realiza nenhuma filtragem a não ser a busca das palavras-chave. Este método é muito utilizado também em *sites* de busca na *Internet*, como o Google ([www.google.com](http://www.google.com)), por exemplo. Os três primeiros métodos de recomendação citados são considerados não-personalizados, pois todos os usuários que receberem recomendações de um desses três métodos receberão a mesma recomendação.

- **Baseado em atributos**

Um sistema baseado em atributos recomenda produtos levando em consideração suas características. Para uma pessoa que escuta muitos *CDs* de uma banda específica, por exemplo, um lançamento de *CD* desta banda seria uma possível recomendação do sistema para ele. Outro exemplo é a oferta de desconto no valor de um *CD* de um gênero musical determinado, enquanto o usuário navega numa seção do *site* que vende esse gênero.

- **Correlação item-para-item**

O método de correlação item-para-item tenta identificar produtos que, de alguma forma, têm relação com produtos que o usuário demonstrou interesse no seu histórico de interação com a loja virtual. Um exemplo desse tipo de recomendação é a venda casada. Por exemplo, a oferta de um par de meias quando o usuário compra um tênis. Aplicações podem inspecionar itens dentro do carrinho de compras para identificar essas associações. Esse tipo de recomendação tem como objetivo principal aumentar as vendas e oferecer um serviço diferenciado aos usuários. No momento em que o usuário está realizando alguma compra, o *site* tenta convencer-lhe a comprar mais um produto.

- **Correlação usuário-para-usuário**

O método de correlação usuário-para-usuário leva em conta a colaboração de usuários que têm preferências parecidas, visto que, estes provavelmente manterão a coerência em seus produtos comprados. Esta técnica recomenda produtos de acordo com a similaridade do usuário-alvo com outros usuários do *site* de comércio eletrônico.

### **2.3.6 Descobrindo Hábitos de Consumo**

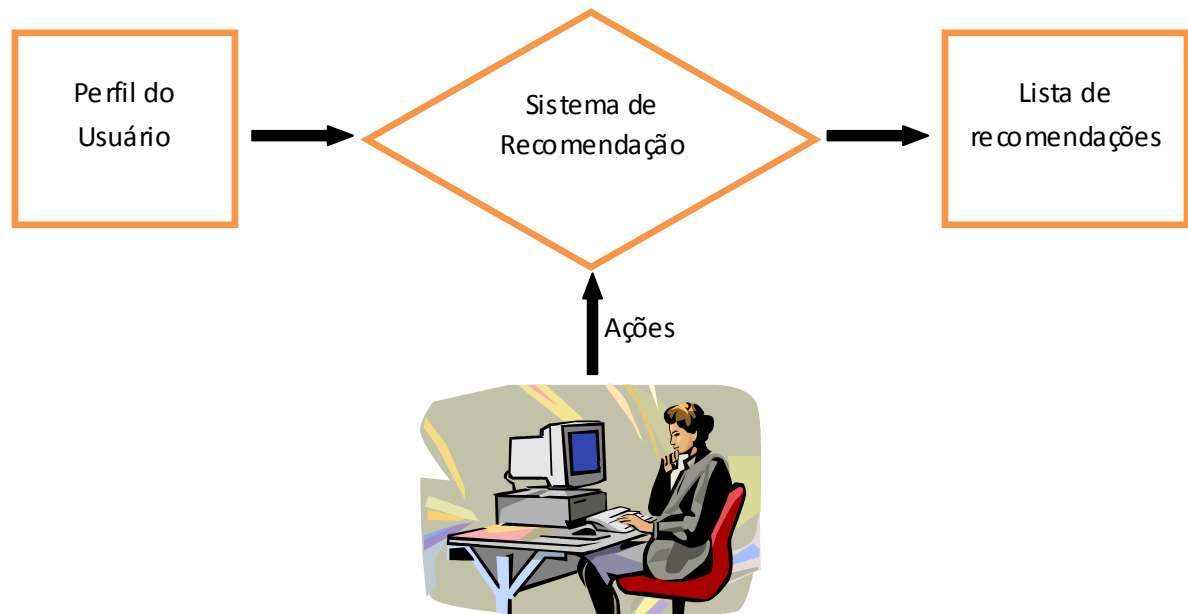
Sistemas de recomendação podem ser utilizados para que se possa conhecer melhor os hábitos de consumo e interesses dos usuários, e como este tipo de conhecimento pode ser empregado para personalizar o relacionamento entre a empresa e seus consumidores [2].

Para que o perfil do usuário seja construído, é necessário haver uma ou mais interações do usuário com o *site* até que as preferências sejam identificadas. Existem basicamente duas formas de extrair o perfil do usuário, a implícita e a explícita [7]:

- Forma implícita: As informações implícitas são coletadas de maneira que o usuário não perceba que o sistema está coletando informações sobre o mesmo. Essas informações são coletadas durante a navegação do usuário, tais como: tempo de visita a uma página, itens visitados e/ou selecionados, número de acessos por espaço de tempo (diário, semanal, quinzenal), entre outras. Outra forma implícita são os produtos adquiridos, armazenando-se o movimento do histórico de compras do usuário;
- Forma explícita: as informações explícitas são fornecidas intencionalmente pelos usuários. Nelas o usuário expressa de alguma forma se gosta ou não do produto quantitativamente, por exemplo, fornecendo notas aos produtos. Esse tipo de informação é considerado mais confiável, já que o usuário é quem fornece a informação, mas o custo desse tipo de procedimento é o esforço do usuário que nem sempre está disposto a colaborar com mais informações para o sistema. Geralmente são coletadas informações como notas atribuídas para um produto. Essas notas são obtidas por interfaces simples, por exemplo, com possibilidades de notas variando de 1 a 5, ou indicações binárias simples informando se gosta ou não gosta de uma lista de produtos indicados.

### **2.3.7. Gerando Recomendações**

Para que sejam geradas recomendações, deve-se considerar o funcionamento de um sistema de recomendação, onde são extraídas as informações do perfil do usuário através de suas ações de navegação para que o Sistema de Recomendação gere uma listagem de itens recomendáveis para o usuário como ilustra a Figura 4.



**Figura 4:** Funcionamento de um Sistema de Recomendação [7]

Uma lista de recomendações é gerada após o usuário praticar alguma atividade no *site*. Pode-se citar entre essas atividades a entrada no *site* e adicionar algum produto na cesta de compras ou solicitar recomendações explícitas ao sistema. Quando o usuário solicita recomendações de forma explícita, utiliza-se o perfil do usuário para gerá-las. Em geral, a lista de recomendações é composta por produtos mais similares ao perfil do usuário [7].



## Capítulo 3

### Projeto do Software e Aspectos Técnicos

Esse Capítulo descreve o projeto do software desenvolvido bem como os aspectos técnicos envolvidos no processo.

A Seção 3.1 apresenta uma breve contextualização sobre as decisões acerca da implementação do processo definido para o funcionamento do Sistema de Recomendação. A metodologia de desenvolvimento é apresentada na Seção 3.2. A Seção 3.3 descreve a especificação dos requisitos funcionais, análise e projeto do sistema. Os demais requisitos do sistema estão descritos na Seção 3.4. A modelagem dos casos de uso é apresentada na Seção 3.5. As descrições textuais dos casos de uso do sistema estão definidas na Seção 3.6. O modelo de classes conceituais é apresentado na Seção 3.7. O glossário com a apresentação dos termos utilizados no projeto foi descrito pela Seção 3.8. A Seção 3.9 denota as visões de classes participantes dos casos de uso. O modelo de iterações é representado na Seção 3.10. A descrição textual dos contratos das operações que abrangem o sistema é apresentada na Seção 3.11. A Seção 3.12 exibe os diagramas de seqüência para cada cenário relevante dos casos de uso. A seção 3.13 descreve o projeto de banco de dados. O projeto lógico de banco de dados é apresentado na Seção 3.14. A Seção 3.15 apresenta o projeto físico do banco de dados. Finalmente a Seção 3.16 descreve o projeto da interface gráfica, seguido pela Seção 3.17 que descreve a hierarquia das telas e mapa de navegação.

#### 3.1 Contextualização

Após estudar Sistemas de Recomendação, foi tomada a decisão de utilizar a técnica de extração de informações implícita. A técnica envolve o processo de seleção e apresentação de itens com atributos semelhantes ao item selecionado pelo usuário, ou seja, quando o são apresentados itens parecidos com o item procurado pelo usuário. Isso ocorre quando o usuário seleciona um item, sendo oferecido pela interface ou por uma lista resultante de uma consulta, desta forma o sistema busca e define itens de maior similaridade, para assim apresentar as recomendações para ele.

As recomendações são apresentadas ao usuário, quando o mesmo as solicitar, através de um *link* “Clique aqui se deseja receber recomendações!”, *pop-ups* (nova janela de tamanho inferior à janela do *site* que é ativada pelo sistema) e/ou através de *banners* (que são distribuídos no *layout* do *site*).

As seções a seguir descrevem informações relativas ao desenvolvimento de Sistemas de Informação para a *Web* e as tecnologias que estarão presentes no *site* proposto neste TCC, utilizando Sistemas de Recomendação.

### **3.2 Metodologia de desenvolvimento**

Tendo como ponto de partida artigos de internet e consulta a referências bibliográficas, a abordagem utilizada no desenvolvimento do sistema é o Paradigma de Orientação a Objetos.

Através da linguagem PHP 5 [19], além das tecnologias e ferramentas de desenvolvimento como: Framework Yii [20], MySQL [21] como sistema de gerenciador de banco de dados (SGBD), jQuery [22], CSS [23] e o Astah [24]. O Adobe Fireworks [25] é utilizado para a construção das interfaces que compõem o sistema; o JavaScript é utilizado para efetuar validações e tratamento de entrada de dados por parte do usuário; o CSS é aplicado com o objetivo de organizar o layout e a aparência das interfaces do sistema; o framework Yii é utilizado para minimizar a manutenção das classes de domínio sendo também responsável pela camada de persistência da aplicação com o respectivo mapeamento objeto relacional; o MySQL é o responsável pelo gerenciamento de dados contidos nas tabelas da aplicação; o jQuery é responsável por dinamizar o uso de controles de formulários; o Astah é responsável pela construção dos modelos que compõem o sistema.

### **3.3 Especificação, Análise e Projeto do Sistema**

Este capítulo aborda os aspectos de especificação, análise e projeto de sistema. A descrição detalhada e a especificação dos requisitos funcionais e não funcionais são

compreendidas na seção 3.3.1. Na Seção 3.3.2 contém a representação do modelo de casos de uso do sistema onde são especificados o diagrama de casos de uso, a descrição textual dos atores e o detalhamento dos casos de uso do sistema. O modelo de classes em conjunto com o diagrama de classes de negócio, dicionário das classes e a visão das classes participantes são apresentados na Seção 3.3.3. Na Seção 3.3.4 denota o modelo de interações contendo a descrição textual de contratos e diagramas de sequência. Na Seção 3.3.5 estão descritos o projeto lógico e físico de banco de dados do sistema. Por final, a Seção 3.3.6, apresenta o projeto de interface gráfica que compreende a hierarquia de telas, mapa de navegação e padronização de controles visuais.

### **3.3.1 Especificação dos requisitos**

Os requisitos funcionais são declarações de serviços que o sistema deve fornecer como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações. Seus requisitos não funcionais compreendem restrições sobre serviços ou funções oferecidos pelo sistema tais como restrições de timing, restrições sobre o processo de desenvolvimento, padrões, etc... Ou seja, fixam restrições sobre como os requisitos funcionais serão implementados.

### **3.3.2 Especificação dos requisitos funcionais do sistema**

Os requisitos funcionais foram definidos para a construção do sistema:

- i. O sistema deve permitir que o corretor cadastre, altere, consulte e exclua os imóveis os quais selecionou;
- ii. O sistema deve permitir que o corretor cadastre, altere, consulte e exclua os incorporadores aos quais se afiliou;
- iii. O corretor deve poder obter as estatísticas das consultas realizadas no site em determinado período;
- iv. O sistema deve permitir que um cliente selecione imóveis e os envie para um endereço de e-mail;
- v. O sistema deve exibir os imóveis similares sempre que o cliente visualizar um determinado produto;

### 3.4 Especificação dos demais requisitos do sistema

- i. A interface gráfica deve propiciar a fácil usabilidade por parte do usuário. Não fornecendo: recursos de áudio, tela sensível ao toque e teclas de atalho. Faz uso de teclado, mouse.
- ii. O tempo de resposta ao usuário não deve exceder 1 (um) segundo. A aplicação foi desenvolvida para ser acessada através de qualquer computador conectado à internet.

### 3.5 Modelo de Casos de Uso

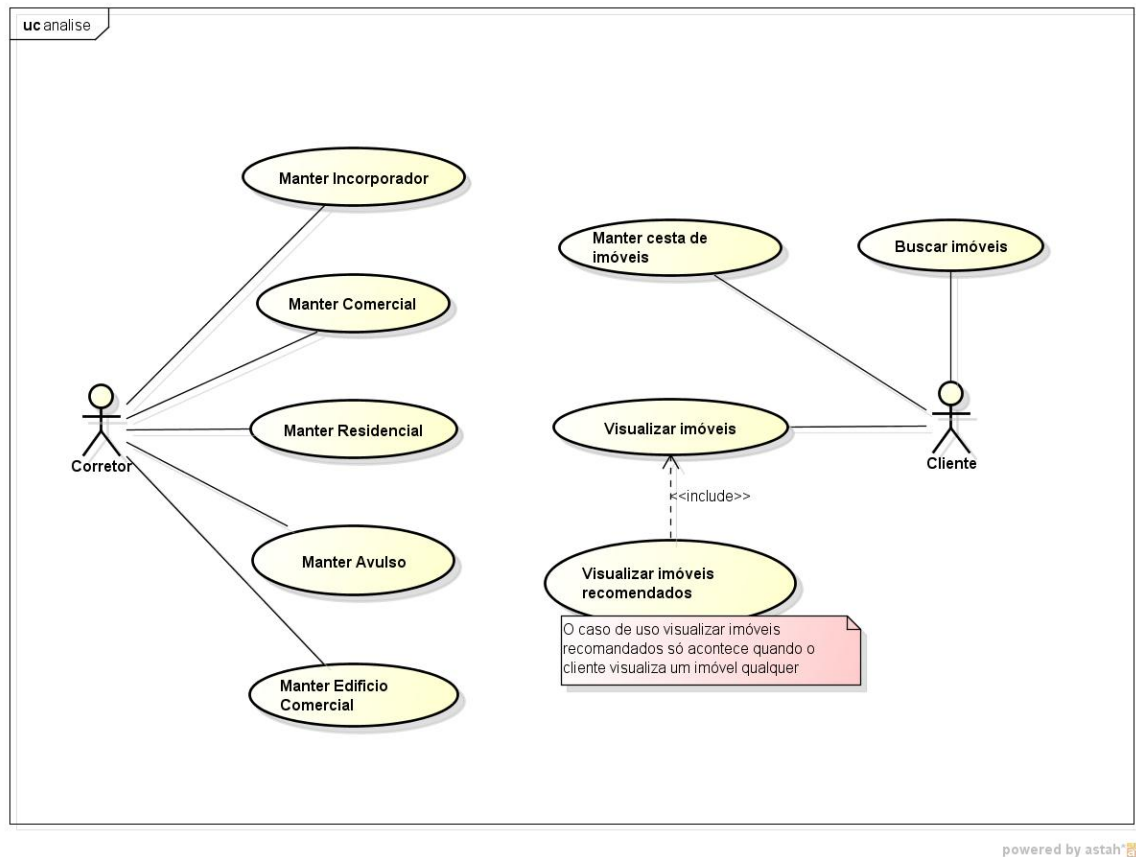
Um modelo de caso de uso tem como objetivo descrever como diferentes tipos de usuários interagem com o sistema para resolver um problema. Como tal, ele descreve as metas dos usuários, as interações entre os usuários e o sistema, bem como o comportamento necessário do sistema para satisfazer estas metas, em resumo, atua um unificador em todo o desenvolvimento do sistema. É usado como a principal especificação dos requisitos funcionais para o sistema, servindo como base para a análise e o design, como uma entrada para o planejamento da iteração, como base para a definição de casos de teste e como base para a documentação dos usuários.

Os elementos de modelo mais importantes são: casos de uso, atores e as relações entre eles. A próxima seção apresenta o diagrama de casos de uso do nosso sistema que corresponde à maneira como a linguagem de modelagem unificada (UML) representa a modelagem de casos de uso.

#### 3.5.1 Diagrama de Casos de Uso

Os Diagramas de Caso de Uso descrevem relacionamentos e dependências entre um grupo de *Caso de Uso* e os Atores participantes no processo. Diagramas de Caso de Uso são feitos para facilitar a comunicação com os futuros usuários do sistema, e são especialmente úteis para determinar os recursos necessários que o sistema deve ter. São produzidos no

intuito de representar a visão externa de alto nível do sistema, de forma a demonstrar os elementos externos que se relacionam com as funcionalidades que fazem parte do sistema.



**Figura 5:** Diagrama de casos de uso

### 3.5.2 Descrições textuais dos atores

A descrição textual dos atores que compõem o sistema está descrita em detalhes a seguir nesta seção.

- i. Corretor: É o responsável por manter o cadastro dos imóveis. Realiza as funcionalidades principais da aplicação;
- ii. Cliente: É o responsável pela realização das funcionalidades de busca de imóveis, gerenciamento da cesta de imóveis, e outras.

### 3.6 Descrições textuais dos casos de uso

Nesta seção estão definidos em detalhes os casos de uso que compreendem o sistema.

#### **Manter Avulso**

##### Sumário

Este caso de uso permite que o Corretor mantenha as informações sobre os imóveis avulsos no sistema. Isso abrange inclusão, exclusão, alteração e visualização dos imóveis avulsos no sistema.

##### Ator Primário

Corretor.

##### Fluxo Principal

1. O Corretor acessa o sistema e seleciona a opção do menu *Avulsos*;
2. O sistema redireciona o Corretor para uma tela com a lista dos imóveis avulsos adicionados recentemente, um menu com a operação de inclusão e um formulário de busca.

##### Fluxo Alternativo (2) : Incluir Imóvel Avulso

O Corretor seleciona a operação *Incluir Novo Avulso* no menu de operações:

- a. O sistema redireciona o Corretor para uma tela com um formulário de cadastro;
- b. O Corretor insere os dados e os envia para o sistema;
- c. O sistema valida os dados, se os dados forem válidos o sistema criará um novo registro para o imóvel avulso, caso contrário retornará ao item *b* informando os erros de preenchimento ao Corretor;
- d. O sistema exibe a tela de visualização do imóvel avulso.

##### Fluxo Alternativo (2) : Alterar Imóvel Avulso

O Corretor seleciona um imóvel da lista de exibição:

- a. O sistema exibe a tela de visualização do imóvel avulso;
- b. O Corretor seleciona a opção *Editar Ficha Técnica*;
- c. O sistema exibe uma tela com o formulário de edição dos dados para o Corretor;

- d. O Corretor alterar as informações desejadas e envia o formulário ao sistema;
- e. O sistema valida os dados, se os dados forem válidos o sistema registrará as modificações realizadas pelo Corretor, caso contrário retornará ao item *c* informando os erros de preenchimento;
- f. O sistema exibe a tela de visualização do imóvel avulso.

#### Fluxo Alternativo (2) : Excluir Imóvel Avulso

O Corretor seleciona um imóvel da lista de exibição:

- a. O sistema exibe a tela de visualização do imóvel avulso;
- b. O Corretor seleciona a opção Excluir Imóvel Avulso;
- c. O sistema requer confirmação do Corretor para prosseguir com a exclusão;
- d. O Corretor confirma a exclusão;
- e. O sistema remove o imóvel avulso do sistema.

### **Manter Residenciais**

#### Sumário

Este caso de uso permite que o Corretor mantenha as informações sobre os conjuntos residenciais no sistema. Isso abrange inclusão, exclusão, alteração e visualização.

#### Ator Primário

Corretor.

#### Fluxo Principal

- 3. O Corretor acessa o sistema e seleciona a opção do menu *Residenciais*;
- 4. O sistema redireciona o Corretor para uma tela com a lista dos conjuntos residenciais adicionados recentemente, um menu com a operação de inclusão e um formulário de busca.

#### Fluxo Alternativo (2) : Incluir Residencial

O Corretor seleciona a operação *Incluir Residencial* no menu de operações:

- a. O sistema redireciona o Corretor para uma tela com um formulário de cadastro;
- b. O Corretor insere os dados e os envia para o sistema;

- c. O sistema valida os dados, se os dados forem válidos o sistema criará um novo registro para o residencial, caso contrário retomará ao item *b* informando os erros de preenchimento ao Corretor;
- d. O sistema exibe a tela de visualização do residencial.

#### Fluxo Alternativo (2) : Alterar Residencial

O Corretor seleciona um residencial da lista de exibição :

- a. O sistema exibe a tela de visualização do residencial;
- b. O Corretor seleciona a opção *Editar Ficha Técnica*;
- c. O sistema exibe uma tela com o formulário de edição dos dados para o Corretor;
- d. O Corretor alterar as informações desejadas e envia o formulário ao sistema;
- e. O sistema valida os dados, se os dados forem válidos o sistema registrará as modificações realizadas pelo Corretor, caso contrário retomará ao item *c* informando os erros de preenchimento;
- f. O sistema exibe a tela de visualização do residencial.

#### Fluxo Alternativo (2) : Excluir Residencial

O Corretor seleciona um residencial da lista de exibição :

- a. O sistema exibe a tela de visualização do residencial;
- b. O Corretor seleciona a opção Excluir Residencial;
- c. O sistema requer confirmação do Corretor para prosseguir com a exclusão;
- d. O Corretor confirma a exclusão;
- e. O sistema remove o residencial do sistema.

### **Manter Comercial**

#### Sumário

Este caso de uso permite que o Corretor mantenha as informações sobre um imóvel comercial no sistema. Isso abrange inclusão, exclusão, alteração e visualização.

#### Ator Primário

Corretor.

#### Fluxo Principal



1. O Corretor acessa o sistema e seleciona a opção do menu *Comerciais*;
2. O sistema redireciona o Corretor para uma tela com a lista dos imóveis comerciais adicionados recentemente, um menu com a operação de inclusão e um formulário de busca.

#### Fluxo Alternativo (2) : Incluir Comercial

O Corretor seleciona a operação *Incluir Comercial* no menu de operações:

- a. O sistema redireciona o Corretor para uma tela com um formulário de cadastro;
- b. O Corretor insere os dados e os envia para o sistema;
- c. O sistema valida os dados, se os dados forem válidos o sistema criará um novo registro para o imóvel comercial, caso contrário retornará ao item *b* informando os erros de preenchimento ao Corretor;
- d. O sistema exibe a tela de visualização do imóvel comercial.

#### Fluxo Alternativo (2) : Alterar Comercial

O Corretor seleciona um imóvel comercial da lista de exibição:

- a. O sistema exibe a tela de visualização do imóvel comercial;
- b. O Corretor seleciona a opção *Editar Ficha Técnica*;
- c. O sistema exibe uma tela com o formulário de edição dos dados para o Corretor;
- d. O Corretor alterar as informações desejadas e envia o formulário ao sistema;
- e. O sistema valida os dados, se os dados forem válidos o sistema registrará as modificações realizadas pelo Corretor, caso contrário retornará ao item *c* informando os erros de preenchimento;
- f. O sistema exibe a tela de visualização do imóvel comercial.

#### Fluxo Alternativo (2) : Excluir Comercial

O Corretor seleciona um imóvel comercial da lista de exibição:

- a. O sistema exibe a tela de visualização do imóvel comercial;
- b. O Corretor seleciona a opção *Excluir Comercial*;
- c. O sistema requer confirmação do Corretor para prosseguir com a exclusão;
- d. O Corretor confirma a exclusão;
- e. O sistema remove o imóvel comercial.

## **Manter Edifício Comercial**

### Sumário

Este caso de uso permite que o Corretor mantenha as informações sobre um edifício comercial no sistema. Isso abrange inclusão, exclusão, alteração e visualização.

### Ator Primário

Corretor.

### Fluxo Principal

1. O Corretor acessa o sistema e seleciona a opção do menu *Edifícios Comerciais*;
2. O sistema redireciona o Corretor para uma tela com a lista dos edifícios comerciais adicionados recentemente, um menu com a operação de inclusão e um formulário de busca.

### Fluxo Alternativo (2) : Incluir Edifício Comercial

O Corretor seleciona a operação *Incluir Edifício Comercial* no menu de operações:

- a. O sistema redireciona o Corretor para uma tela com um formulário de cadastro;
- b. O Corretor insere os dados e os envia para o sistema;
- c. O sistema valida os dados, se os dados forem válidos o sistema criará um novo registro para o edifício comercial, caso contrário retornará ao item *b* informando os erros de preenchimento ao Corretor;
- d. O sistema exibe a tela de visualização do imóvel comercial.

### Fluxo Alternativo (2) : Alterar Edifício Comercial

O Corretor seleciona um edifício comercial da lista de exibição:

- a. O sistema exibe a tela de visualização do edifício comercial;
- b. O Corretor seleciona a opção *Editar Ficha Técnica*;
- c. O sistema exibe uma tela com o formulário de edição dos dados para o Corretor;
- d. O Corretor alterar as informações desejadas e envia o formulário ao sistema;
- e. O sistema valida os dados, se os dados forem válidos o sistema registrará as modificações realizadas pelo Corretor, caso contrário retornará ao item *c* informando os erros de preenchimento;

- f. O sistema exibe a tela de visualização do edifício comercial.

#### Fluxo Alternativo (2) : Excluir Edifício Comercial

O Corretor seleciona um edifício comercial da lista de exibição :

- a. O sistema exibe a tela de visualização do edifício comercial;
- b. O Corretor seleciona a opção Excluir Edifício Comercial;
- c. O sistema requer confirmação do Corretor para prosseguir com a exclusão;
- d. O Corretor confirma a exclusão;
- e. O sistema remove o edifício comercial.

### **Manter Incorporador**

#### Sumário

Este caso de uso permite que o Corretor mantenha as informações sobre os incorporadores no sistema. Isso abrange inclusão, exclusão, alteração e visualização.

#### Ator Primário

Corretor.

#### Fluxo Principal

1. O Corretor acessa o sistema e seleciona a opção do menu *Incorporadores*;
2. O sistema redireciona o Corretor para uma tela com a lista dos incorporadores adicionados recentemente, um menu com a operação de inclusão e um formulário de busca.

#### Fluxo Alternativo (2) : Incluir Incorporador

O Corretor seleciona a operação *Incluir Incorporador* no menu de operações:

- a. O sistema redireciona o Corretor para uma tela com um formulário de cadastro;
- b. O Corretor insere os dados e os envia para o sistema;
- c. O sistema valida os dados, se os dados forem válidos o sistema criará um novo registro para o incorporador, caso contrário retornará ao item *b* informando os erros de preenchimento ao Corretor;
- d. O sistema exibe a tela de visualização do incorporador.

#### Fluxo Alternativo (2) : Alterar Incorporador

O Corretor seleciona um incorporador da lista de exibição :

- a. O sistema exibe a tela de visualização do incorporador;
- b. O Corretor seleciona a opção *Editar*;
- c. O sistema exibe uma tela com o formulário de edição dos dados para o Corretor;
- d. O Corretor alterar as informações desejadas e envia o formulário ao sistema;
- e. O sistema valida os dados, se os dados forem válidos o sistema registrará as modificações realizadas pelo Corretor, caso contrário retomarão ao item c informando os erros de preenchimento;
- f. O sistema exibe a tela de visualização do incorporador.

#### Fluxo Alternativo (2) : Excluir Incorporador

O Corretor seleciona um incorporador da lista de exibição:

- a. O sistema exibe a tela de visualização do incorporador;
- b. O Corretor seleciona a opção Excluir;
- c. O sistema requer confirmação do Corretor para prosseguir com a exclusão;
- d. O Corretor confirma a exclusão;
- e. O sistema remove o incorporador.

### **Manter Cesta de Imóveis**

#### Sumário

Este caso de uso permite que o Cliente armazene as informações sobre os imóveis que visualizou no site em uma cesta de compra. Isso abrange incluir, listar e excluir e os imóveis selecionados.

#### Ator Primário

Cliente

#### Fluxo Principal

1. O Cliente acessa o site e seleciona um produto da lista de produtos oferecidos
2. O sistema exibe a ficha técnica do imóvel assim como recomenda imóveis com a mesma características
3. O Cliente clica no botão Adicionar a Cesta de Imóveis

4. O sistema adiciona o imóvel a cesta de imóveis e exibe os itens da cesta junto com um formulário que permite o envio da cesta por e-mail
5. O Cliente preenche o e-mail e envia os dados ao sistema
6. O sistema verifica se o endereço é um e-mail válido, caso não seja ele retorna ao item 4 exibindo os erros de preenchimento
7. O sistema envia a lista dos imóveis para o e-mail informado pelo Cliente
8. O caso de uso termina

#### Fluxo Alternativo (5) : Excluir Imóvel da Cesta

- a. O Cliente seleciona a opção Excluir
- b. O sistema requer confirmação do Cliente para prosseguir com a exclusão
- c. O Cliente confirma a exclusão
- d. O sistema remove o imóvel da Cesta de Imóveis

### **Visualizar Imóveis**

#### Sumário

Este caso de uso permite que o Cliente visualize imóveis com características similares as que ele está pesquisando.

#### Ator Primário

Cliente

#### Fluxo Principal

1. O Cliente acessa o site e seleciona um produto da lista de produtos oferecidos;
2. O sistema exibe a ficha técnica do imóvel e uma lista de imóveis com características similares com as do imóvel que ele optou por visualizar;
3. O caso de uso termina.

#### Fluxo Alternativo (2) : Visualizar Imóveis Recomendados

- a. O Cliente seleciona dos imóveis da lista de imóveis recomendados e o Caso de Uso retoma para o passo 2.

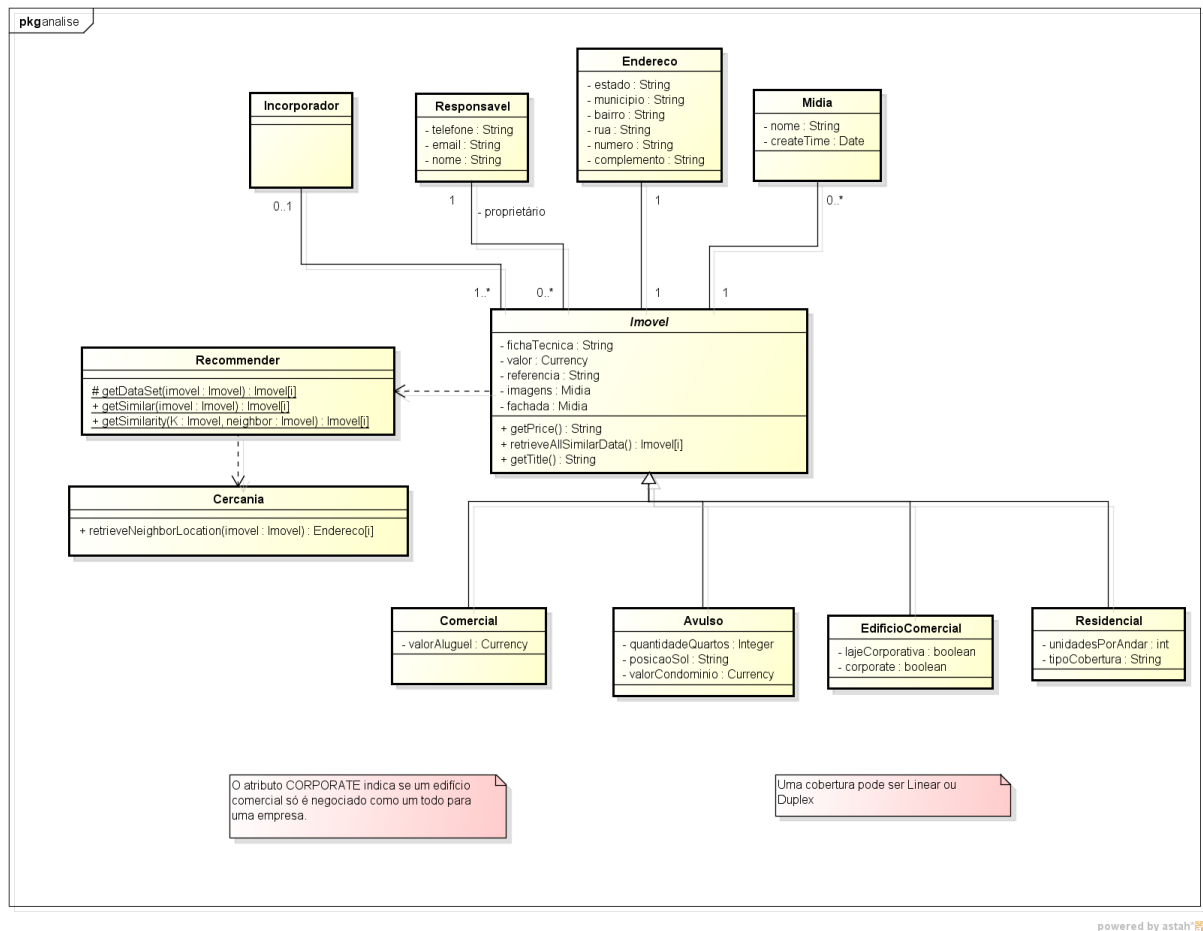
### **3.7 Modelo de Classes**

O modelo de classes tem como objetivo representar a visão interna do sistema, definindo como os objetos do sistema colaboram uns com os outros no intuito de produzir resultados visíveis. Desta forma, o modelo de classes permite compreender a estrutura do sistema para que as funcionalidades sejam reproduzidas. Na próxima seção apresenta o diagrama de classes conceituais do sistema em concordância com a forma definida pela linguagem de modelagem unificada (UML) para representar o modelo de classes.

#### **3.7.1 Diagrama de Classes Conceituais**

O diagrama de classes permite compreender o aspecto estrutural interno do sistema. O diagrama de classes contém as funcionalidades externamente visíveis do sistema, a colaboração entre elementos no aspecto estrutural estático assim como as associações entre os elementos do diagrama [26].

É possível ilustrar internamente os conceitos importantes do sistema em um domínio como classes conceituais e suas associações conforme denotado pela Figura 6.



**Figura 6:** Diagrama de classes conceituais

### 3.8 Glossário

O Glossário tem como objetivo definir os termos relevantes do domínio, alguns deles correspondentes a classes que compõem o sistema. A definição do glossário é apresentada a seguir: O Glossário a seguir, descreve detalhadamente as particularidades que existem nas classes componentes do sistema [26].

1. **Corretor:** Representa o profissional que trabalha com venda de imóveis;
2. **Imóvel:** Representa a generalização dos produtos comercializados pelo corretor;
3. **Avulso:** Representa um imóvel residencial o qual o responsável pode ser uma pessoa física ou uma pessoa jurídica;
4. **Residencial:** É o agregado de apartamentos avulsos. O cliente se interessa pelas características do residencial como um todo. Cabe ao corretor encaminhar o

cliente para uma unidade qualquer que esteja disponível para venda dentro do residencial;

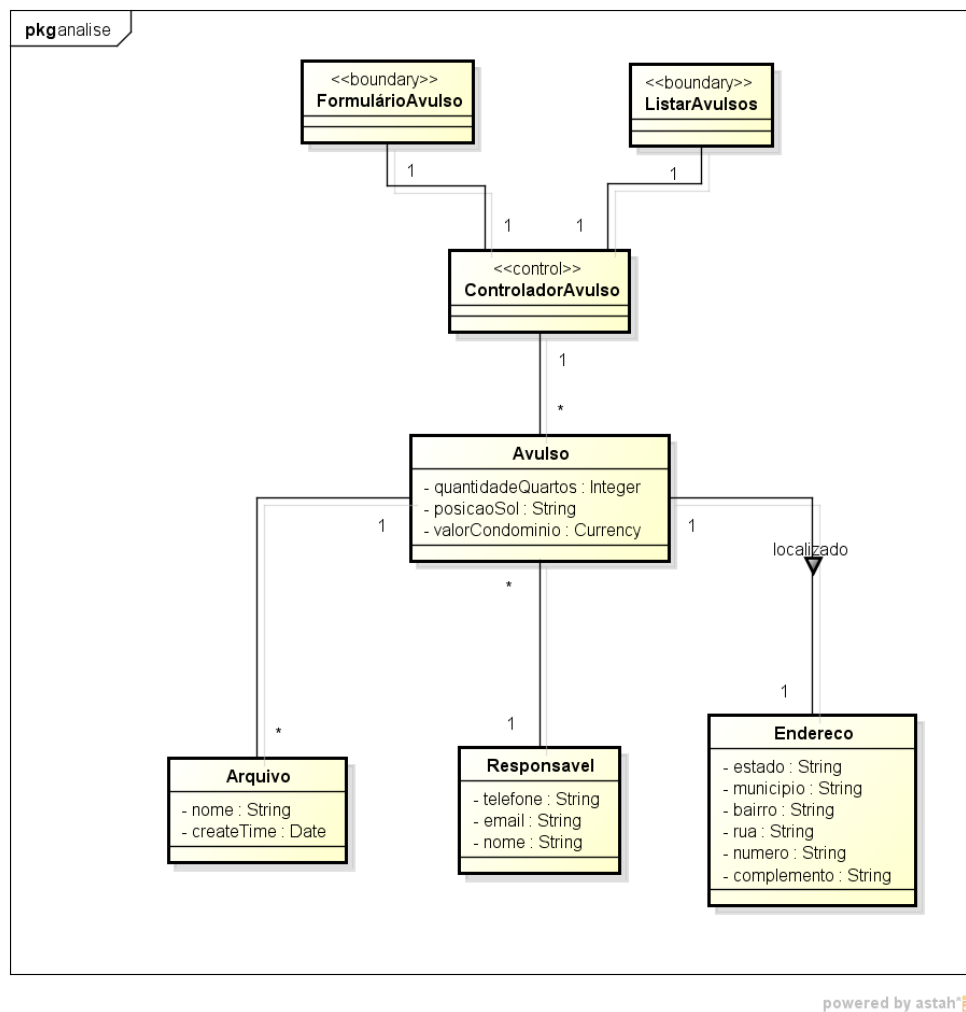
5. **Comercial:** Representa as salas comerciais;
6. **Edifício comercial:** É o agregado de salas comerciais. O edifício comercial pode ser negociado como um todo para um único comprador. Assim como no residencial, o cliente se interessa pelas características do edifício comercial como um todo;
7. **Corporate:** Termo técnico utilizado para indicar quando um edifício comercial só pode ser vendido como um todo para um único cliente;
8. **Endereço:** Posição física do imóvel que indica o estado, município e bairro;
9. **Responsável:** Pessoa física ou jurídica que atende como atual proprietário de um imóvel;
10. **Incorporador:** Empreiteiro responsável por erguer empreendimentos comerciais e residenciais;
11. **Empreendimento:** Construção administrada por um incorporador ou grupo administrativo. As vendas são realizadas nos diversos estados da obra.

### 3.9 Visões de Classes Participantes

A Visão de Classes Participantes (VCP) é um diagrama das classes cujos objetos participam da realização de determinado caso de uso. Tem como principal objetivo representar de forma com menos complexidade, através dos objetos de fronteira, de entidade e de controle, um caso de uso particular reduzindo sua extensão.

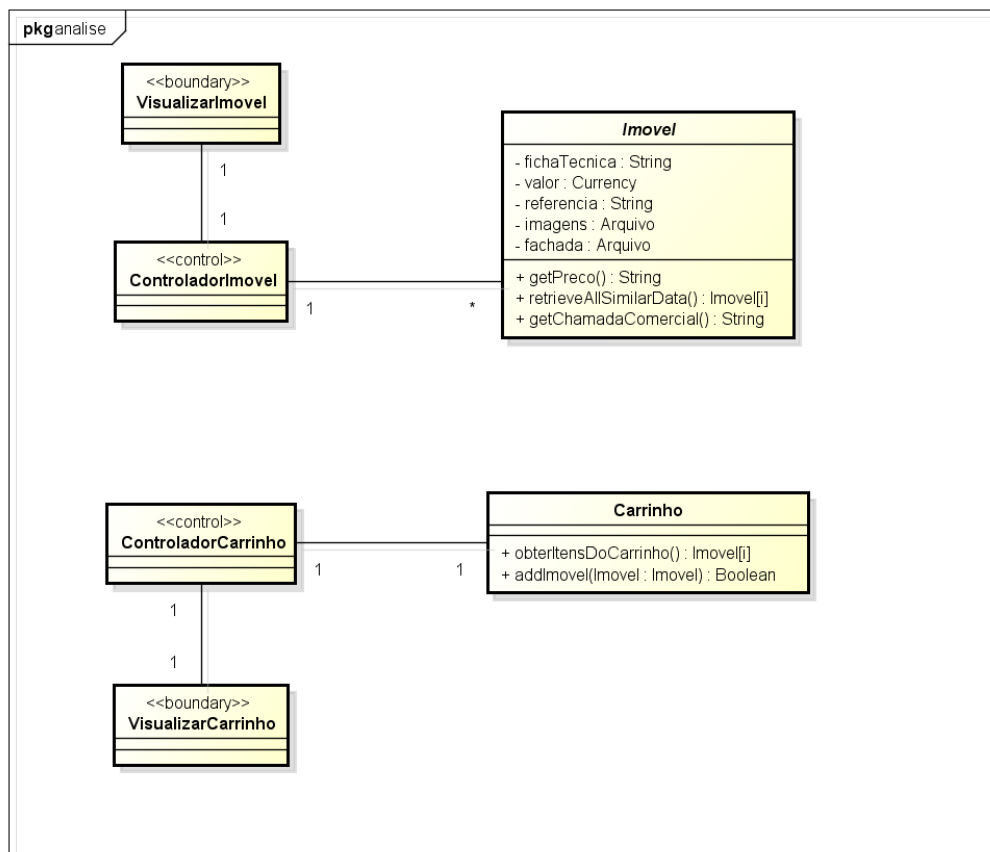
A Figura 7 denota a visão de classes participantes referente ao caso de uso manter Avulso.





**Figura 7: VCP Manter Avulso**

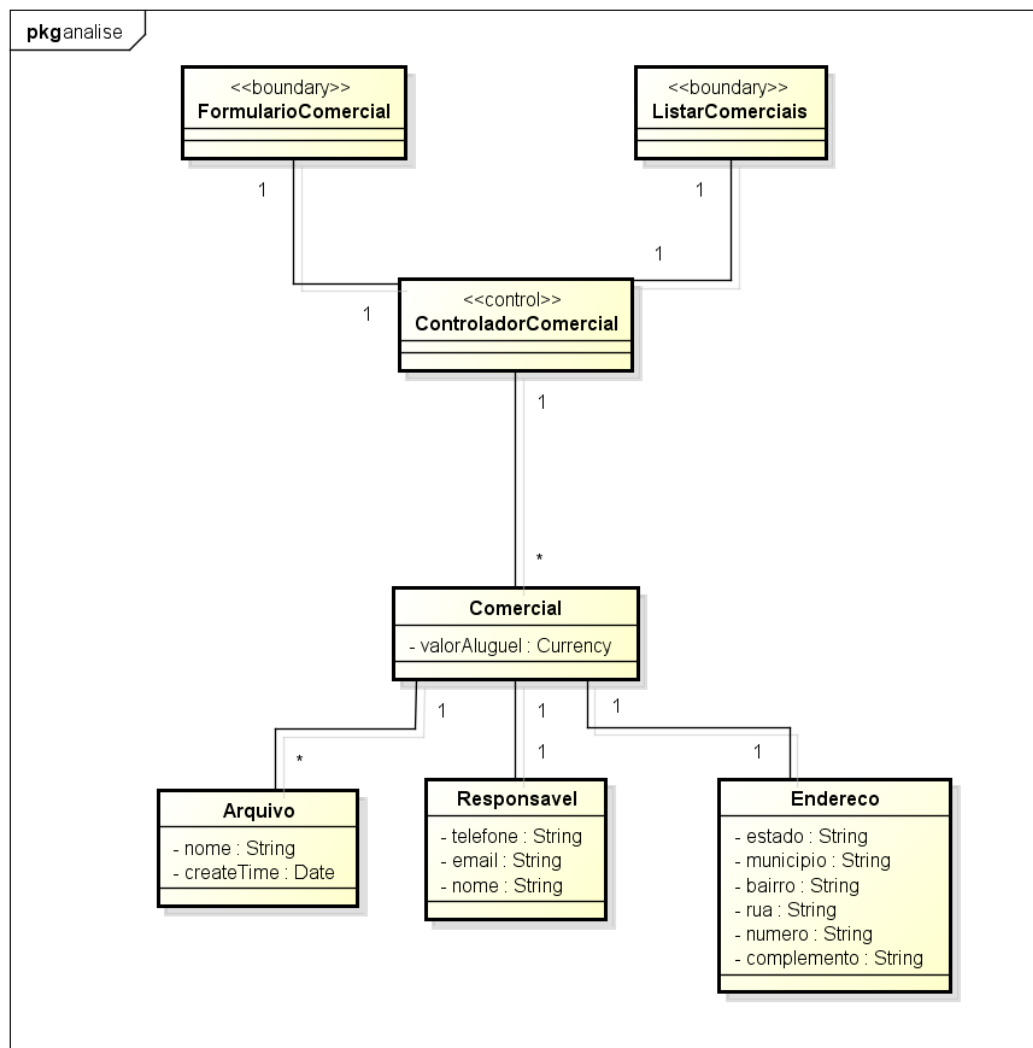
A Figura 8 denota a visão de classes participantes referente ao caso de uso manter Cesta de Imóveis.



powered by astah

**Figura 8:** VCP Manter Cesta de Imóveis

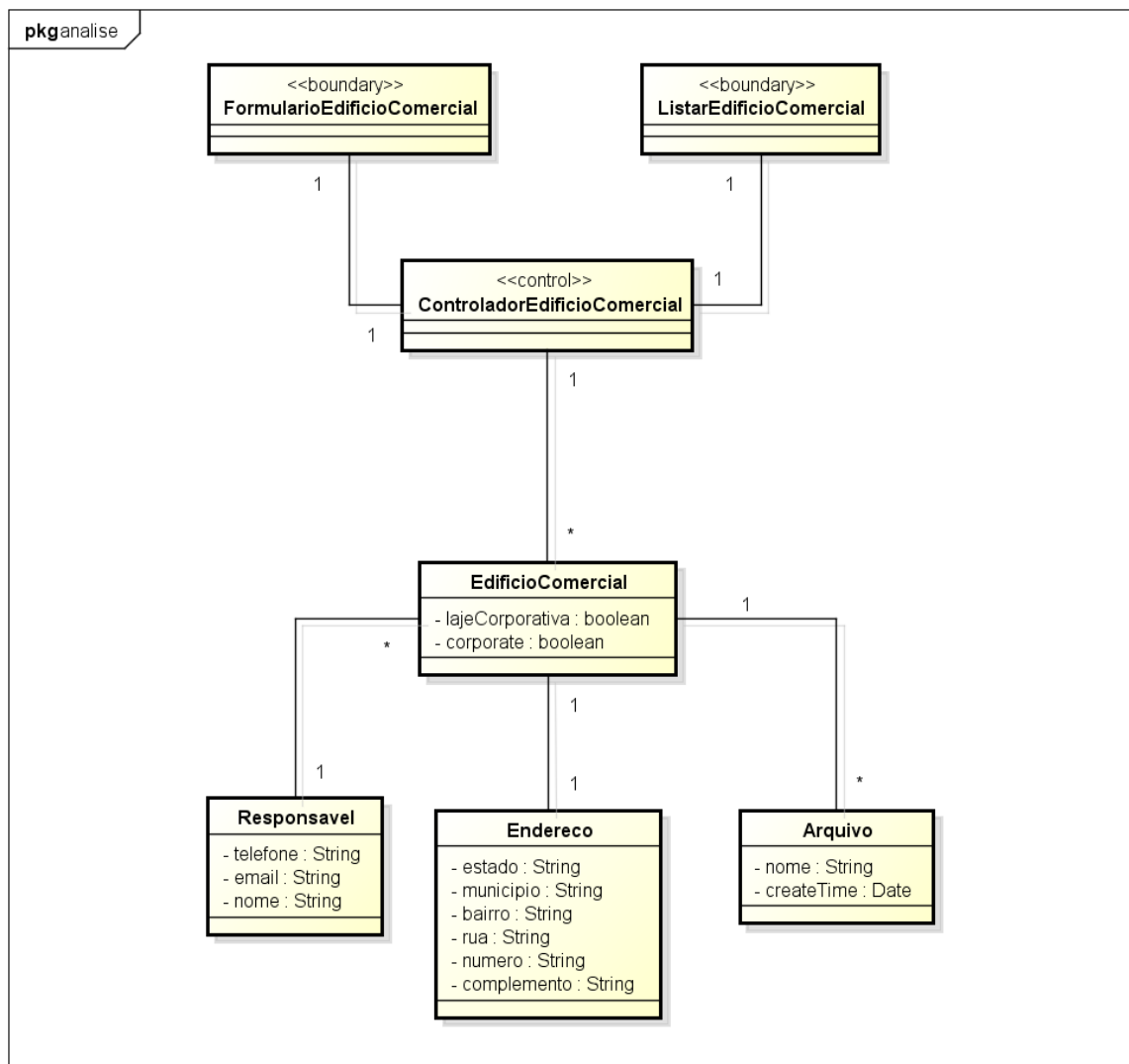
A Figura 9 denota a visão de classes participantes referente ao caso de uso manter Comercial.



powered by astah®

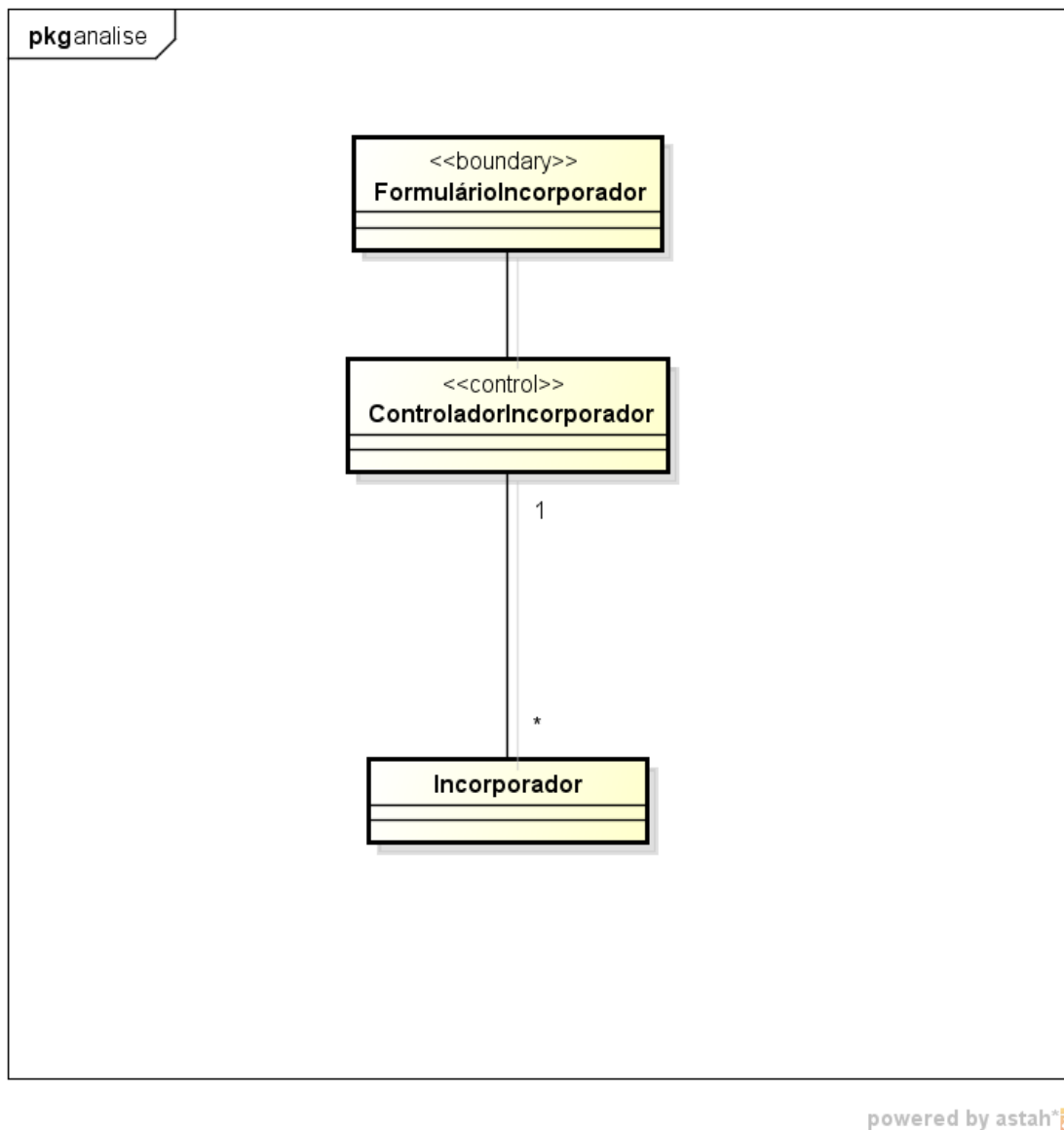
**Figura 9: VCP Manter Comercial**

A Figura 10 denota a visão de classes participantes referente ao caso de uso manter Edifício Comercial.



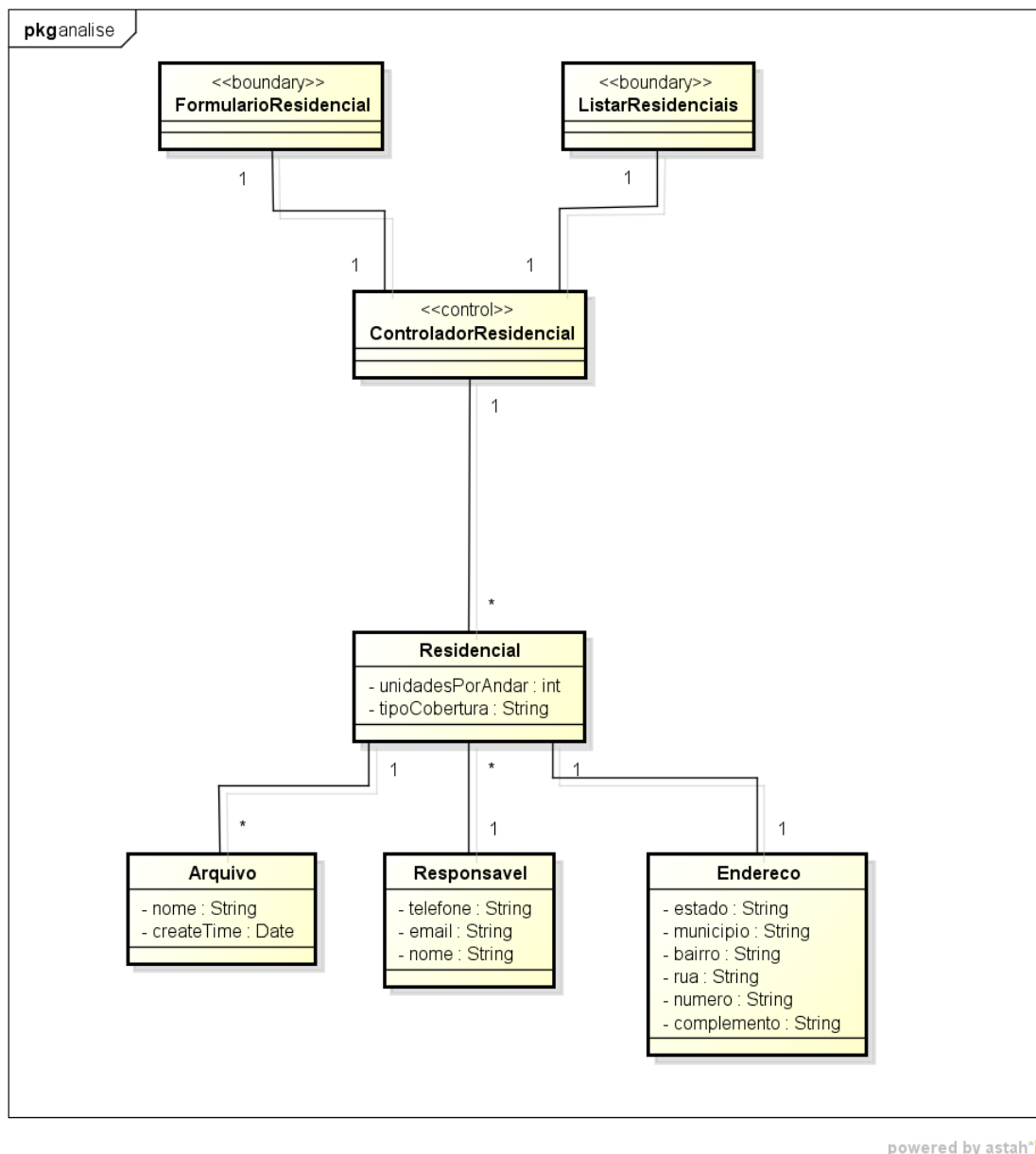
**Figura 10:** VCP Manter Edifício Comercial

A Figura 11 denota a visão de classes participantes referente ao caso de uso manter Incorporador.



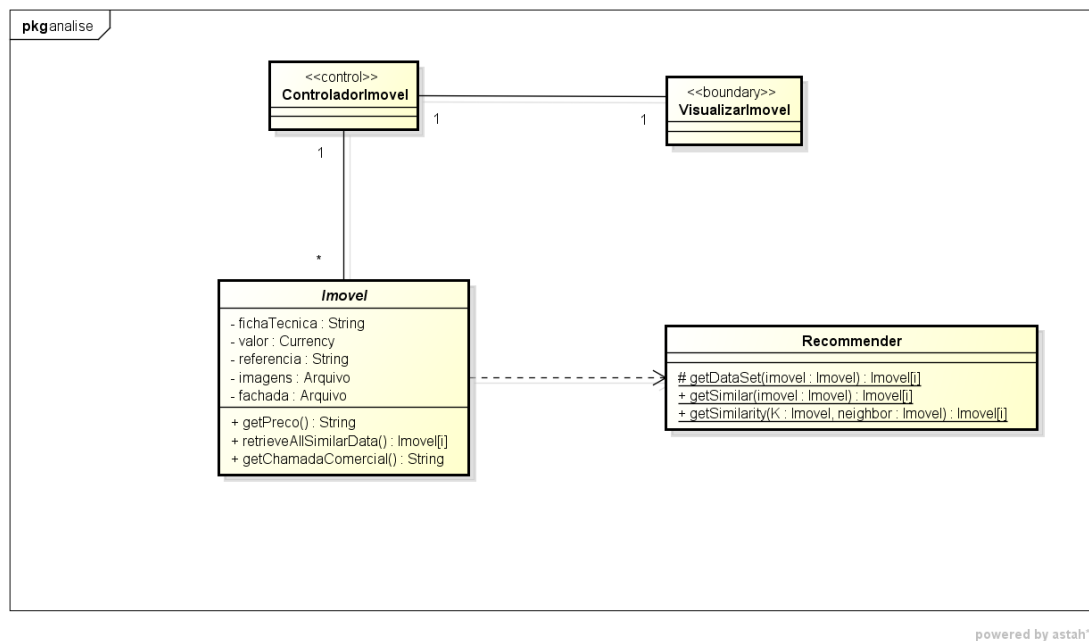
**Figura 11:** VCP Manter Incorporador.

A Figura 12 denota a visão de classes participantes referente ao caso de uso manter Residencial.



**Figura 12:** VCP Manter Residencial.

A Figura 13 denota a visão de classes participantes referente ao caso de uso Visualizar Imóveis Recomendados.



**Figura 13:** VCP Visualizar Imóveis Recomendados.

### 3.10 Modelo de Interações

O modelo de interações tem como serventia ilustrar como os objetos agem internamente para que um ator atinja seu objetivo na realização de um caso de uso. Representa as mensagens trocadas entre os objetos durante a execução de cenários dos casos de uso que compõem o sistema. A linguagem de modelagem unificada (UML) representa os modelos de interação através dos diagramas de sequência.

### 3.11 Descrição textual dos contratos das operações de sistema

O contrato de operações do sistema tem por fim regular as responsabilidades das operações do sistema, retratando os estados de pré e pós-condição. As operações encontram-se disponíveis no modelo de classe.

**Operação:** getDataSet(imovel)

**Responsabilidade:** Retornar um dataset de um determinado tipo de imóvel

**Pré-condição:** N/A

**Pós-condição:** Um conjunto de imóveis da cercania foi gerado

**Operação:** getSimilares(imovel[i])

**Responsabilidade:** Retornar um conjunto de imóveis de imóveis similares ao imóvel parâmetro

**Pré-condição:** N/A

**Pós-condição:** Um conjunto de imóveis presentes na cercania do imóvel informado e similares ao mesmo foi gerado

**Operação:** getSimilaridade(K, vizinho)

**Responsabilidade:** Medir a similaridade entre dois imóveis

**Pré-Condição:** Um dataset de imóveis deve ter sido gerado

**Pós-Condição:** A similaridade entre dois imóveis é expressa em valor numérico

**Operação:** getPreco ()

**Responsabilidade:** Processar o custo de um ou mais imóveis e retomá-lo em um formato amigável ao usuário

**Pré-Condição:** N/A

**Pós-Condição:** O preço de um imóvel é retomado em um formato amigável ao usuário

**Operação:** retrieveAllSimilarData()

**Responsabilidade:** Buscar todos os imóveis similares a partir de um dado exemplo

**Pré-Condição:** N/A

**Pós-Condição:** Um conjunto de imóveis similares é retornado

**Operação:** getChamadaComercial()

**Responsabilidade:** Processar os dados de um imóvel e retomar uma chamada de princípio comercial e amigável ao usuário

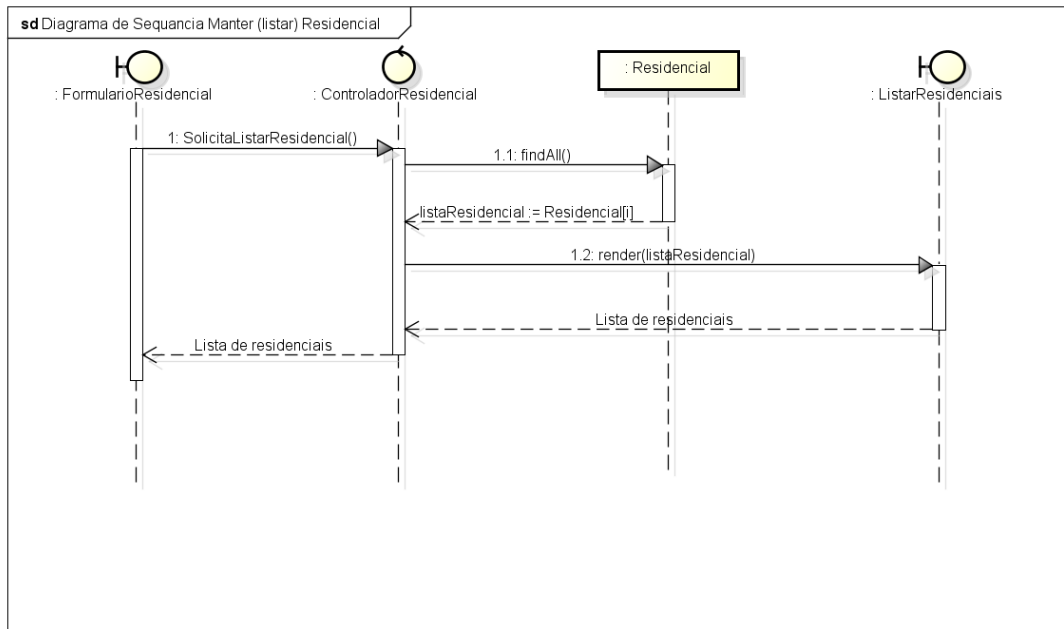
**Pré-Condição:** N/A

**Pós-Condição:** O imóvel é oferecido ao usuário com a chamada comercial



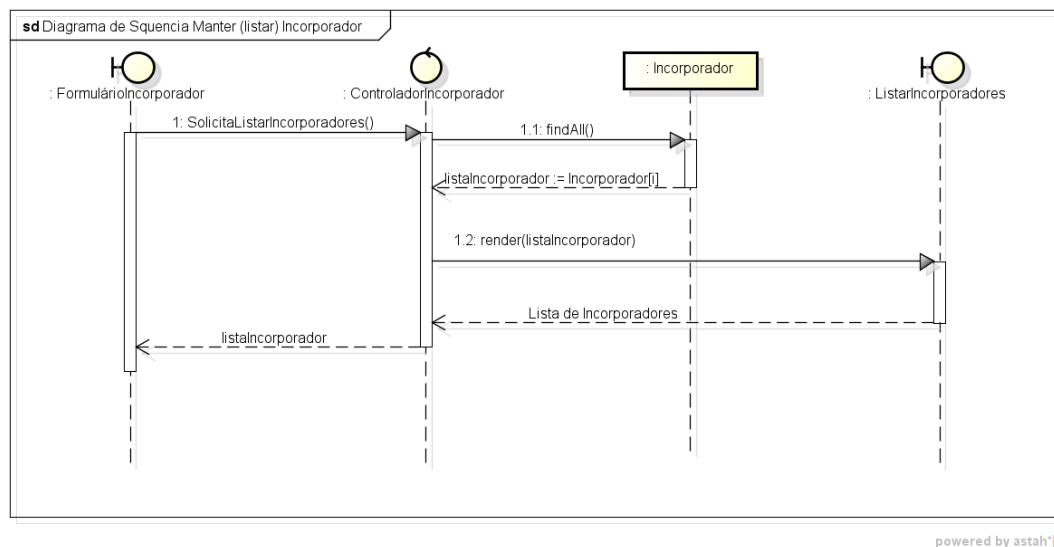
### 3.12 Diagramas de seqüência de cada cenário relevante dos casos de uso

A Figura 14 apresenta o diagrama de seqüência do caso de uso Manter (listar) Residencial.



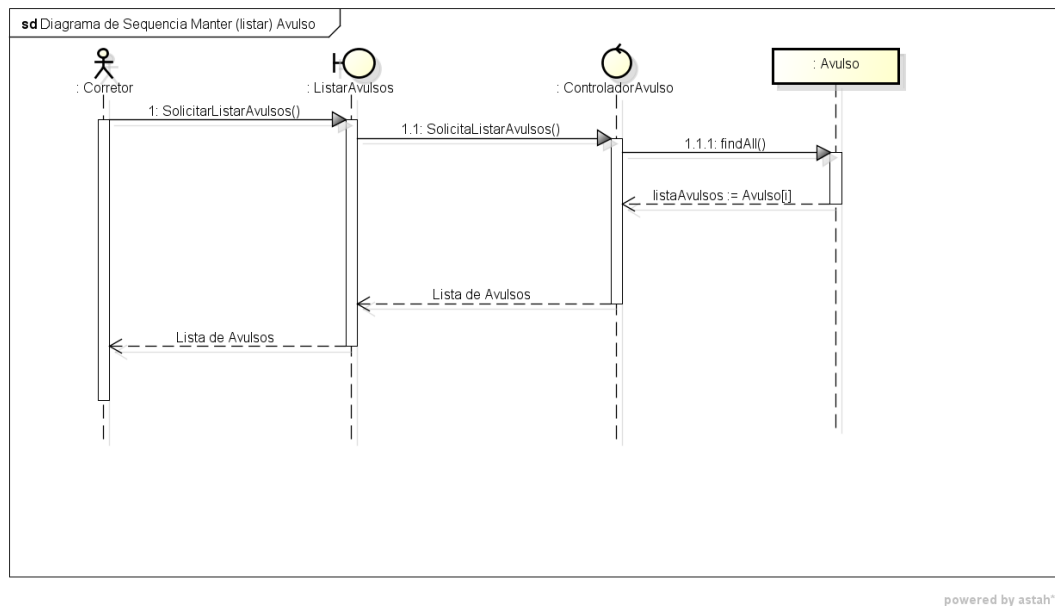
**Figura 14:** Diagrama de Seqüência do caso de uso Manter (listar) Residencial.

A Figura 15 apresenta o diagrama de seqüência do caso de uso Manter (listar) Incorporador.



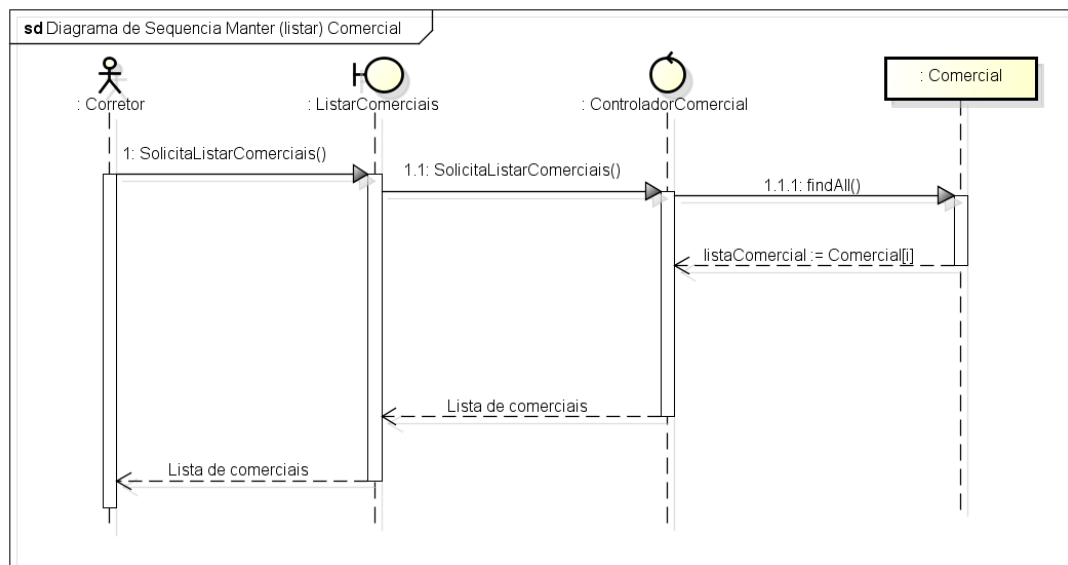
**Figura 15:** Diagrama de Seqüência do caso de uso Manter (listar) Incorporador.

A Figura 16 apresenta o diagrama de seqüência do caso de uso Manter (listar) Avulso.



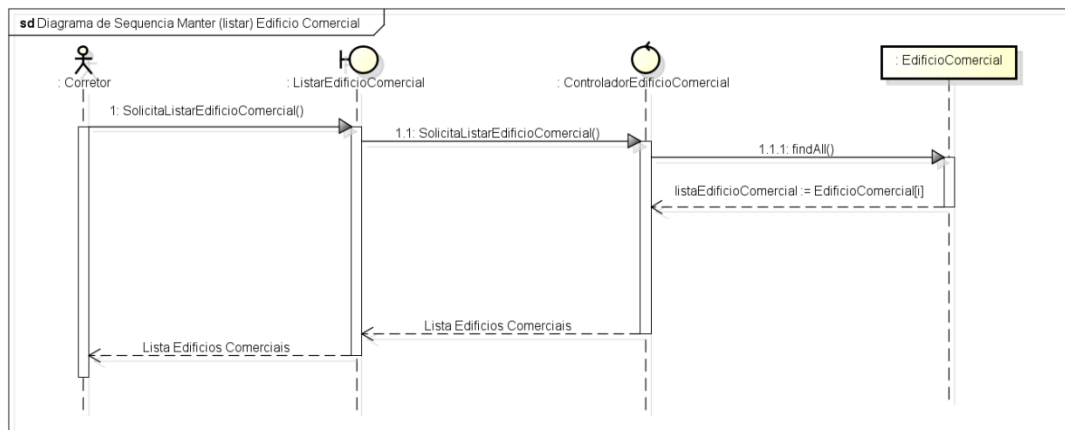
**Figura 16:** Diagrama de Seqüência do caso de uso Manter (listar) Avulso.

A Figura 17 apresenta o diagrama de seqüência do caso de uso Manter (listar) Comercial.



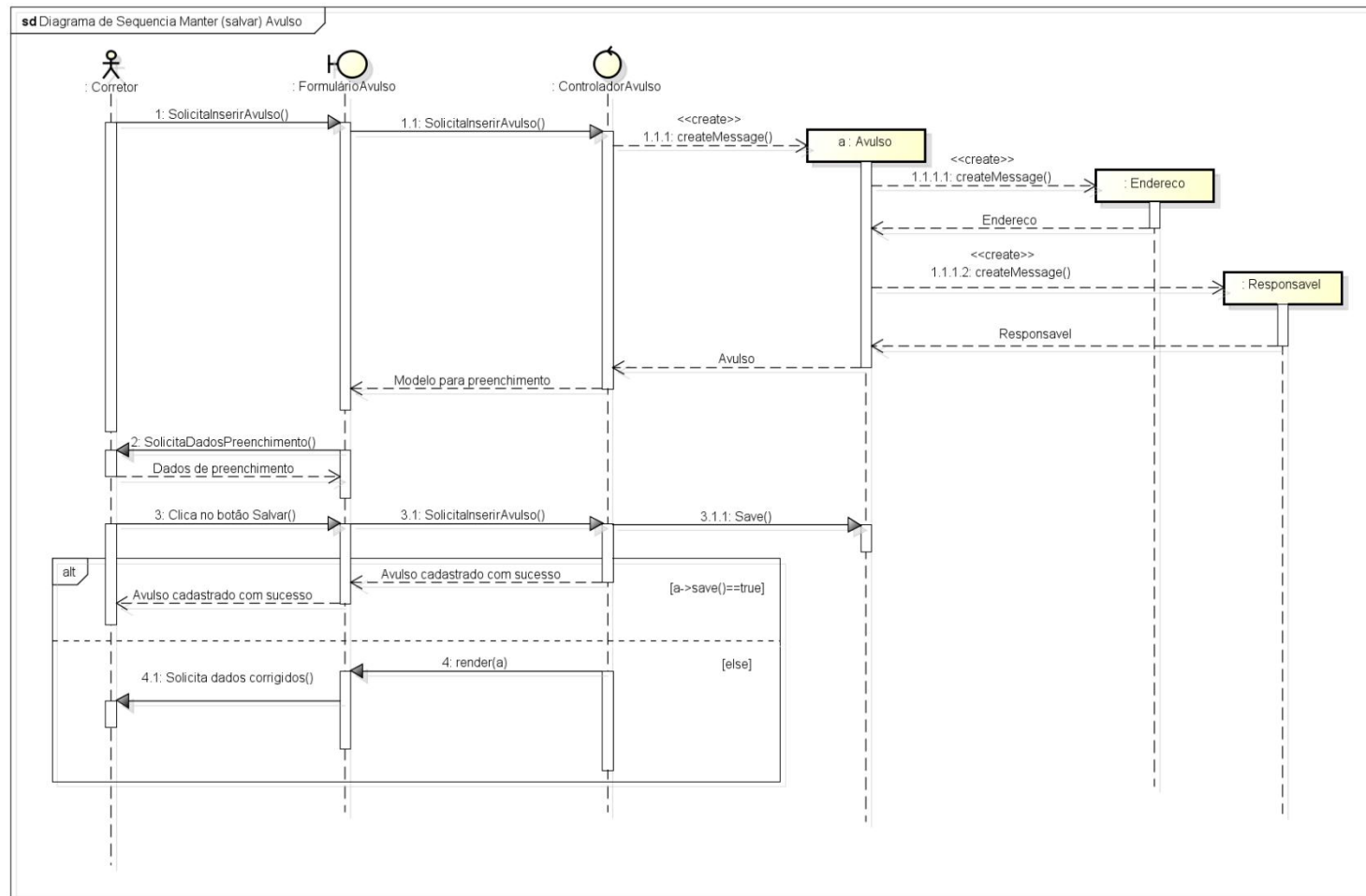
**Figura 17:** Diagrama de Seqüência do caso de uso Manter (listar) Comercial.

A Figura 18 apresenta o diagrama de seqüência do caso de uso Manter (listar) Edifício Comercial.



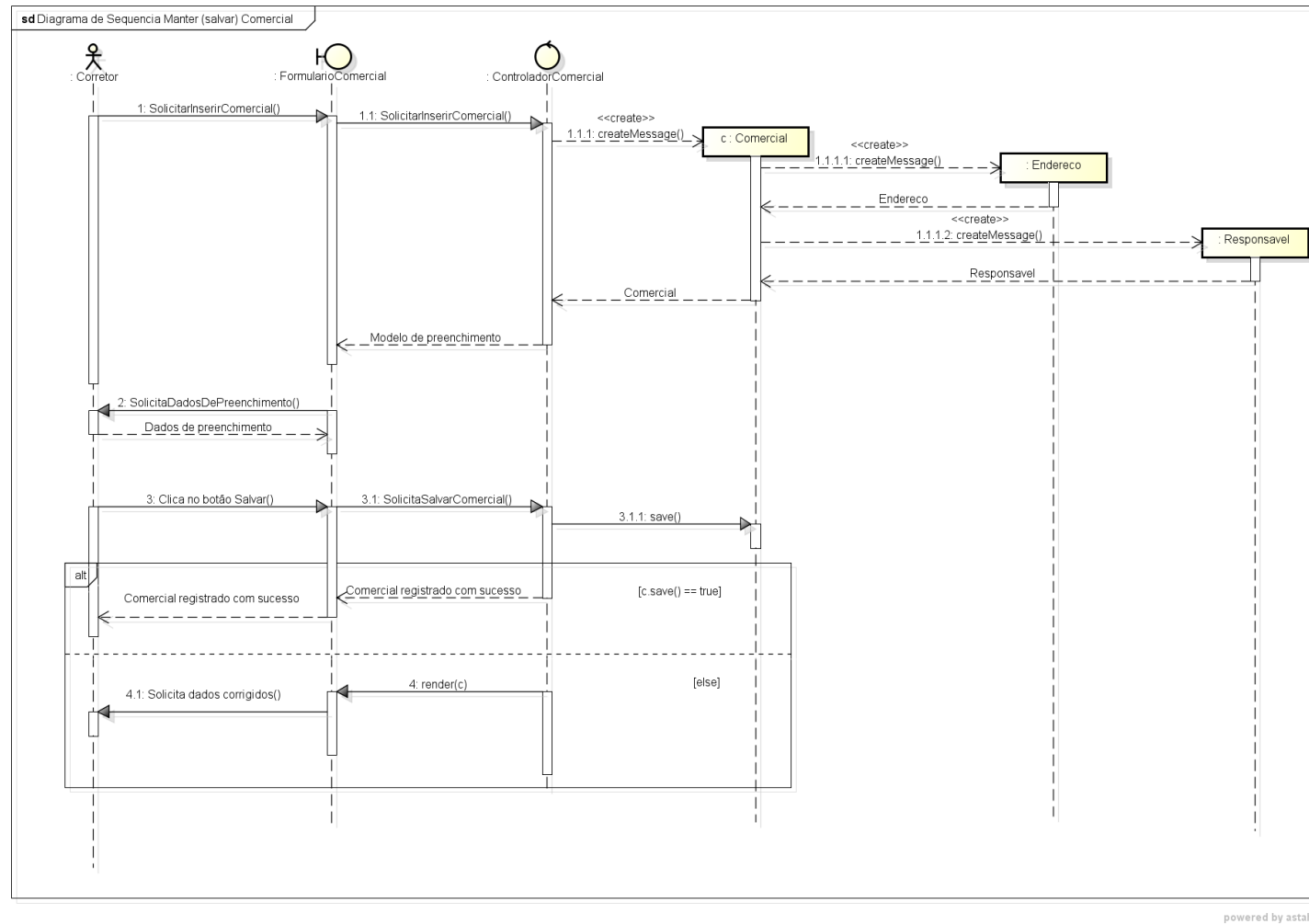
**Figura 18:** Diagrama de Sequência do caso de uso Manter Edifício (listar) Comercial.

A Figura 19 apresenta o diagrama de seqüência do caso de uso Manter (salvar) Avulso.



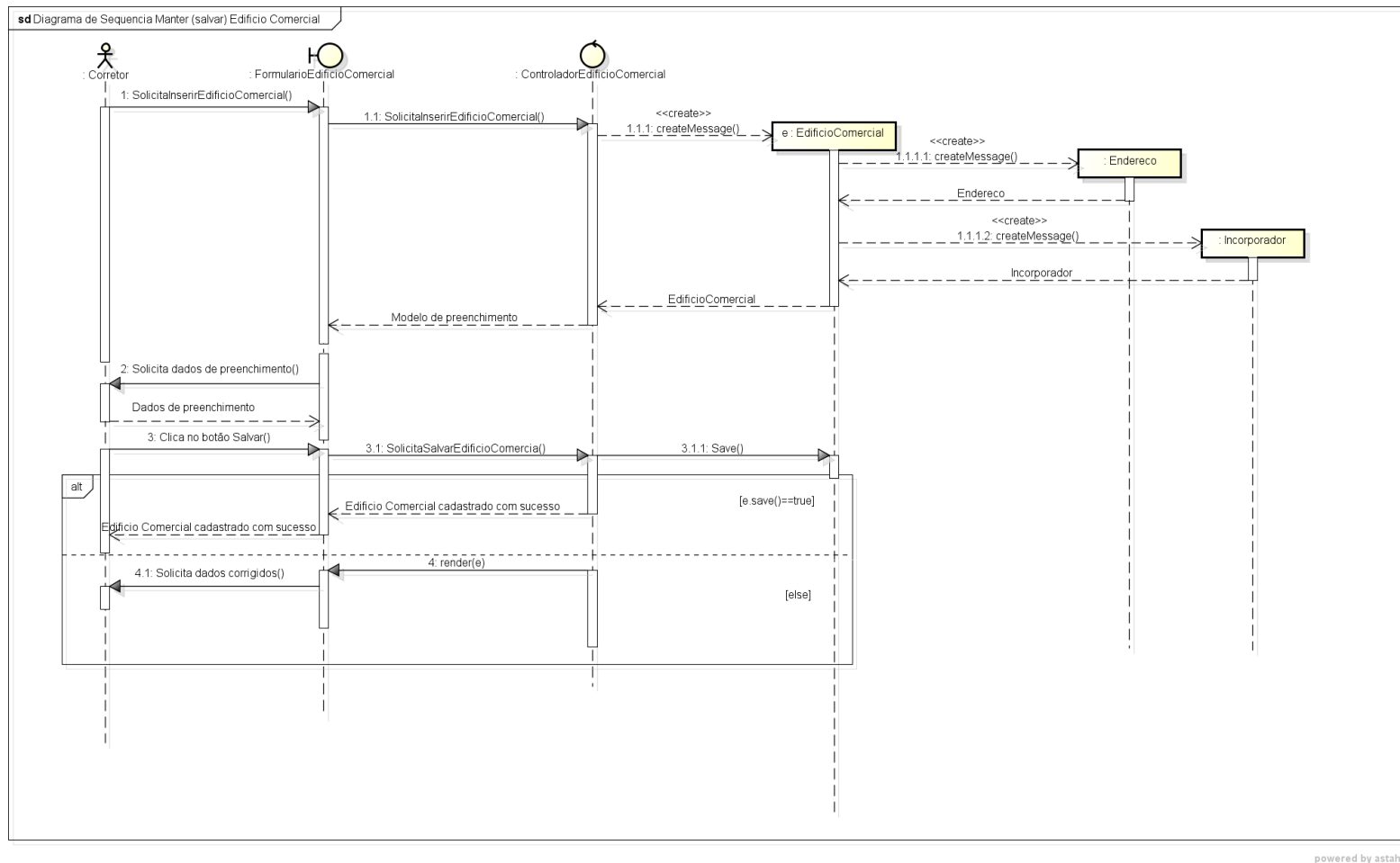
**Figura 19:** Diagrama de Seqüência do caso de uso Manter (salvar) Avulso.

A Figura 20 apresenta o diagrama de seqüência do caso de uso Manter (salvar) Comercial.



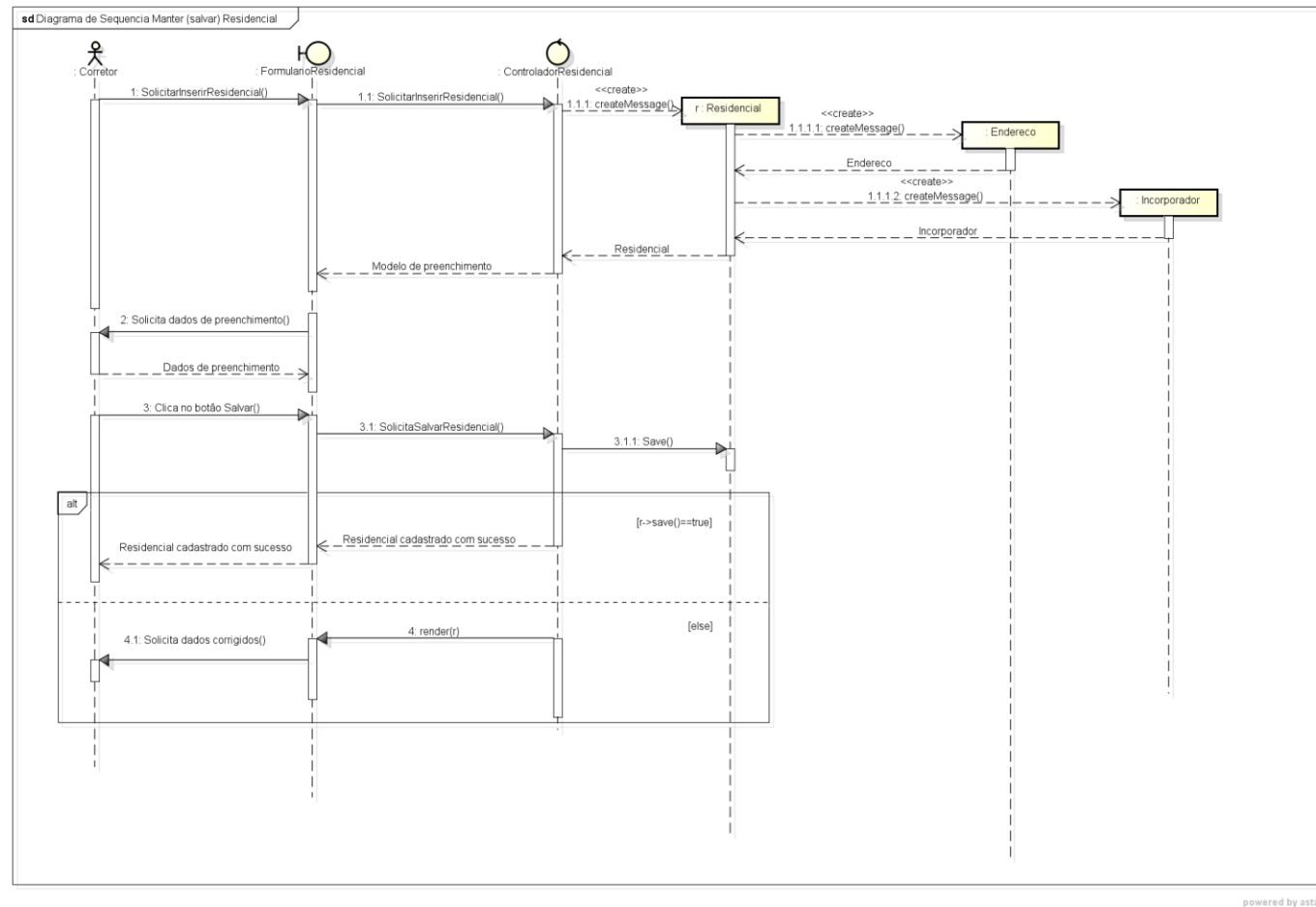
**Figura 20:** Diagrama de Seqüência do caso de uso Manter (salvar) Comercial.

A Figura 21 apresenta o diagrama de seqüência do caso de uso Manter (salvar) Edifício Comercial.



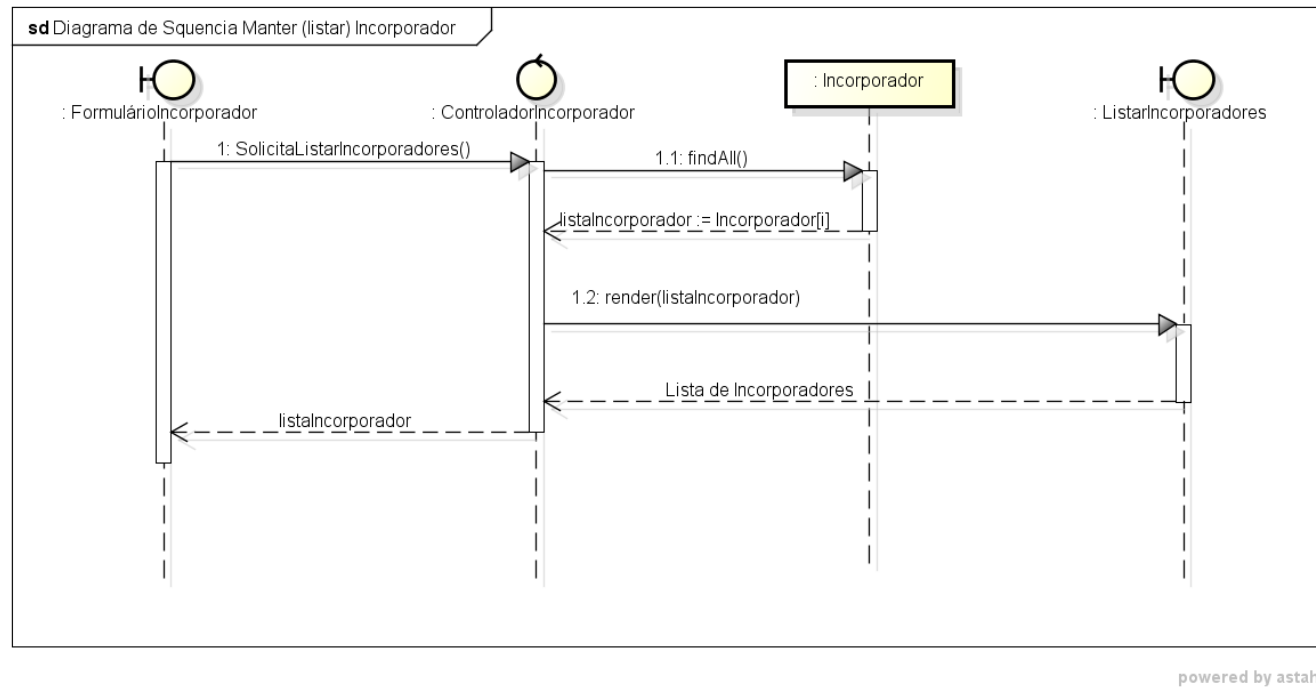
**Figura 21:** Diagrama de Seqüência do caso de uso Manter (salvar) Edifício Comercial.

A Figura 22 apresenta o diagrama de seqüência do caso de uso Manter (salvar) Residencial.



**Figura 22:** Diagrama de Seqüência do caso de uso Manter (salvar) Residencial.

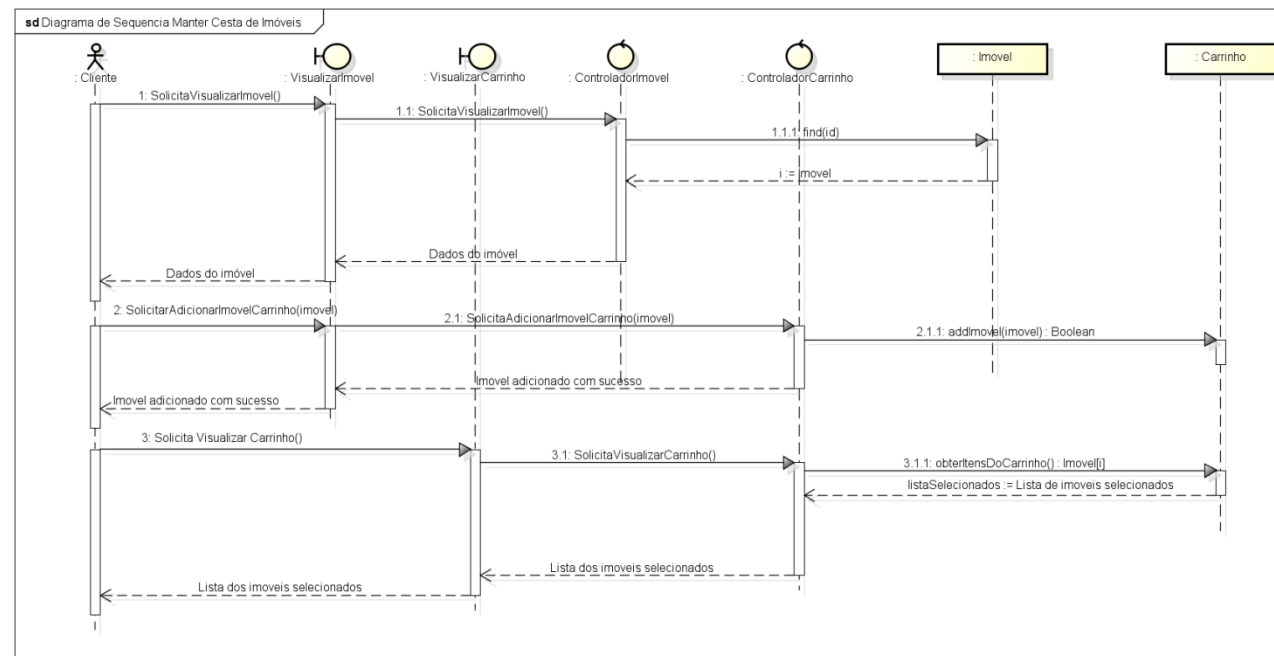
A Figura 23 apresenta o diagrama de seqüência do caso de uso Manter (salvar) Incorporador.



**Figura 23:** Diagrama de Seqüência do caso de uso Manter (salvar) Incorporador.

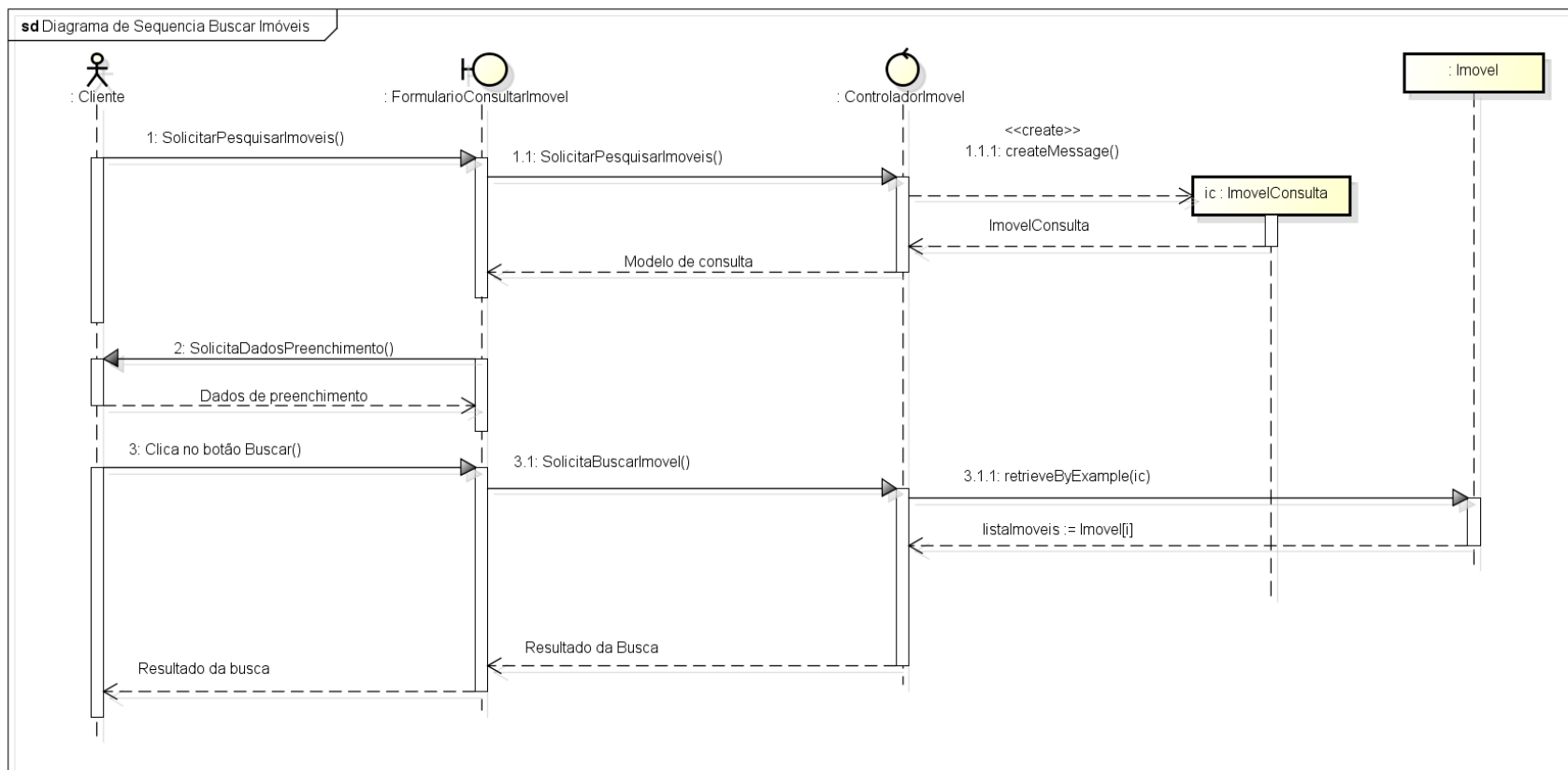
A Figura 24 apresenta o diagrama de seqüência do caso de uso Manter Cesta de Imóveis.





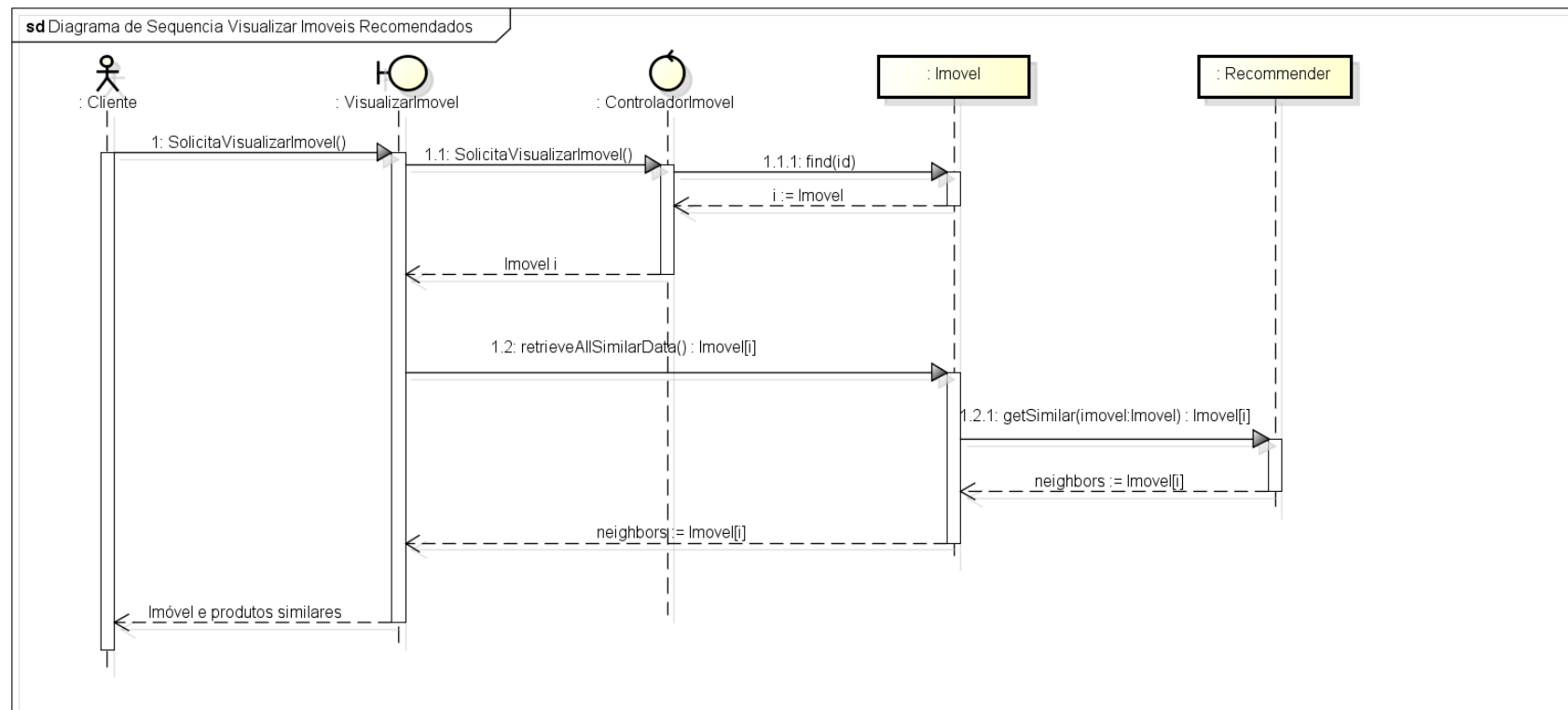
**Figura 24:** Diagrama de Seqüência do caso de uso Manter Cesta de Imóveis.

A Figura 25 apresenta o diagrama de seqüência do caso de uso Buscar Imóveis.



**Figura 25:** Diagrama de Seqüência do caso de uso Buscar Imóveis.

A Figura 26 apresenta o diagrama de seqüência do caso de uso Visualizar Imóveis Recomendados.



powered by astah®

**Figura 26:** Diagrama de Seqüência do caso de Visualizar Imóveis Recomendados.

### 3.13 Projeto de Banco de Dados

O projeto de banco de dados é uma atividade de modelagem que pode ser definida pelos níveis conceitual, lógico e físico.

### 3.14 Projeto Lógico de Banco de Dados

Criar o projeto lógico de banco de dados corresponde a projetar um esquema de banco relacional tendo por base um esquema de projeto conceitual [27]. Cada relação possui um conjunto de atributos. Dentre os elementos deste conjunto, existem os atributos que podem representar chaves primárias, chaves estrangeiras ou ambas. As chaves primárias encontram-se destacadas em **negrito**, as chaves estrangeiras encontram-se formatadas em *itálico*.

AVULSO(**id**, nome, qtd\_quartos, valor, ficha\_tecnica, referencia, deletedDate, estado\_id, municipio\_id, bairro\_id, metragem, create\_time, fachada\_id, posicao\_sol, valor\_condominio, proprietario\_id)

RI1: valor é numérico deve ser decimal com treze dígitos inteiros e dois dígitos fracionários

RI2: referencia deve ser única

RI3: deletedDate deve ser uma data no formato DATETIME

RI4: metragem é numérico

RI5: create\_time deve ser uma data no formato DATE

RI6: posicao\_sol deve ter como domínio um dos valores [manhã|tarde]

RI7: valor\_condominio deve ser numérico com treze dígitos inteiros e dois dígitos fracionários

R8: estado\_id <> null

R9: estado\_id deve corresponder a um registro na relação ENDERECO\_ESTADO

R10: municipio\_id <> null

R11: municipio\_id deve corresponder a um registro na relação ENDERECO\_MUNICIPIO

R12: bairro\_id <> null

R13: bairro\_id deve corresponder a um registro na relação ENDERECO\_BAIRRO

R14: fachada\_id deve corresponder a um registro na relação AVULSO\_ARQUIVO

R15: proprietario \_id deve corresponder a um registro na relação PROPRIETARIO

AVULSO\_ARQUIVO(**id**, arquivo, create\_time, *avulso\_id*, categoria)

RI1: create\_time deve ser uma data no formato DATE

R2: avulso\_id deve corresponder a um registro na relação AVULSO

R3: avulso\_id <> null

CERCANIA(*bairro\_id*, *bairro\_cercania\_id*)

R1: bairro\_id <> null

R2: bairro\_id deve corresponder a um registro na relação ENDERECO\_BAIRRO

R1: bairro\_cercania\_id <> null

R2: bairro\_cercania\_id deve corresponder a um registro na relação ENDERECO\_BAIRRO

COMERCIAL(**id**, nome, *estado\_id*, *municipio\_id*, *bairro\_id*, metragem, ficha\_tecnica, create\_time, valor\_condominio, *fachada\_id*, valor, deletedDate, *proprietario\_id*, *incorporador\_id*, categoria\_comercial, valor\_aluguel)

RI1: metragem é numérico

RI2: create\_time deve ser uma data no formato DATE

RI3: valor\_condominio deve ser decimal com treze dígitos inteiros e dois dígitos fracionários

RI4: valor é numérico deve ser decimal com treze dígitos inteiros e dois dígitos fracionários

RI5: deletedDate deve ser uma data no formato DATETIME

RI6: categoria\_comercial deve ter como domínio um dos valores [sala comercial|terreno|loja|galpão]

RI7: valor\_aluguel deve ser decimal com treze dígitos inteiros e dois dígitos fracionários

R8: estado\_id <> null

R9: estado\_id deve corresponder a um registro na relação ENDERECO\_ESTADO

R10: municipio\_id <> null

R11: municipio\_id deve corresponder a um registro na relação ENDERECO\_MUNICIPIO

R12: bairro\_id <> null

R13: bairro\_id deve corresponder a um registro na relação ENDERECO\_BAIRRO

R14: fachada\_id deve corresponder a um registro na relação COMERCIAL\_ARQUIVO

R15: proprietario\_id deve corresponder a um registro na relação PROPRIETARIO

R16: incorporador\_id deve corresponder a um registro na relação INCORPORADOR

COMERCIAL\_ARQUIVO(**id**, arquivo, create\_time, *comercial\_id*, categoria)

RI1: create\_time deve ser uma data no formato DATE

R2: comercial\_id deve corresponder a um registro na relação COMERCIAL

R3: comercial\_id <> null

EDIFICIO\_COMERCIAL(**id**, ficha\_tecnica, visitas, deletedDate, nome, *estado\_id*, *municipio\_id*, *bairro\_id*, *fachada\_id*, *incorporador\_id*, referencia, preco\_minimo, preco\_maximo, metragem\_minima, metragem\_maxima, status\_obra, laje\_corporativa, corporate, categoria\_comercial)

RI1: visitas é numérico

RI2: deletedDate deve ser uma data no formato DATETIME

RI3: referencia deve ser única

RI4: preco\_minimo deve ser numérico com treze dígitos inteiros e dois dígitos fracionários

RI4: preco\_maximo deve ser numérico com treze dígitos inteiros e dois dígitos fracionários

RI5: metragem\_maxima é numérico

RI6: metragem\_minima é numérico

RI7: status\_obra deve ter como domínio um dos valores [construção|lançamento|pronto|pré-lançamento]

RI8: laje\_corporativa deve ser booleano

RI9: corporate deve ser booleano

RI10: categoria\_comercial deve ter como domínio um dos valores [conjunto de salas|prédio estacionamento|shopping|prédio comercial]

R11: estado\_id <> null

R12: estado\_id deve corresponder a um registro na relação ENDERECO\_ESTADO

R13: municipio\_id <> null

R14: `municipio_id` deve corresponder a um registro na relação `ENDERECO_MUNICIPIO`

R15: `bairro_id` <> null

R16: `bairro_id` deve corresponder a um registro na relação `ENDERECO_BAIRRO`

R17: `fachada_id` deve corresponder a um registro na relação `EDIFICIO_COMERCIAL_ARQUIVO`

`EDIFICIO_COMERCIAL_ARQUIVO(id, arquivo, create_time, edificio_comercial_id, categoria)`

R1: `create_time` deve ser uma data no formato DATE

R12: `edificio_comercial_id` deve corresponder a um registro na relação `EDIFICIO_COMERCIAL`

R3: `edificio_comercial_id` <> null

`ENDERECO_BAIRRO(id, nome, estado_id, municipio_id)`

R1: `estado_id` <> null

R2: `estado_id` deve corresponder a um registro na relação `ENDERECO_ESTADO`

R1: `municipio_id` <> null

R2: `municipio_id` deve corresponder a um registro na relação `ENDERECO_MUNICIPIO`

`ENDERECO_ESTADO(id, nome, uf)`

`ENDERECO_MUNICIPIO(id, nome, estado_id)`

R1: `estado_id` <> null

R2: `estado_id` deve corresponder a um registro na relação `ENDERECO_ESTADO`

`INCORPORADOR(id, nome, descricao, imagem, deletedDate)`

R1: `deletedDate` deve ser uma data no formato DATETIME

`PROPRIETARIO(id, nome, telefone, email)`

`REGISTRO_CONSULTA(registro_busca_id, estado, cidade, bairro, data_consulta, faixa_preco, tipo_produto, categoria)`

RI1: data\_consulta deve ser uma data no formato DATE

RESERVA\_IMOVEL(**id**, email, session\_id, data\_de\_envio, imovel\_id, imovel\_categoria)

RI1: data\_de\_envio deve ser uma data no formato DATE

R2: imovel\_categoria deve ter como domínio um dos valores [comercial|edificio comercial|residencial|avulso]

RESIDENCIAL(**id**, incorporador\_id, deletedDate, nome, lancamento, data\_entrega, habitesse, metragem\_minima, metragem\_maxima, qtd\_quartos\_minima, qtd\_quartos\_maxima, ficha\_tecnica, referencia, preco\_minimo, preco\_maximo, visitas, estado\_id, municipio\_id, bairro\_id, fachada\_id, create\_time, unidades\_por\_andar, tipo\_cobertura, status\_obra)

RI1: deletedDate deve ser uma data no formato DATE

R2: lancamento deve ser uma data no formato DATE

R3: data\_entrega deve ser uma data no formato DATE

R4: habitesse deve ser uma data no formato DATE

R5: metragem\_minima deve ser numérico

R6: metragem\_maxima deve ser numérico

R7: qtd\_quartos\_minima deve ser numérico

R8: qtd\_quartos\_maxima deve ser numérico

R9: referencia deve ser único

R10: preco\_minimo deve ser decimal com treze dígitos inteiros e dois dígitos fracionários

R11: preco\_maximo deve ser decimal com treze dígitos inteiros e dois dígitos fracionários

R12: visitas deve ser numérico

R13: unidades\_por\_andar deve ser numérico

R14: tipo\_cobertura deve ter como domínio um dos valores [linear|duplex]

R15: status\_obra deve ter como domínio um dos valores [construção|lançamento|pronto|pré-lançamento]

R16: estado\_id <> null

R17: estado\_id deve corresponder a um registro na relação ENDERECO\_ESTADO

R18: municipio\_id <> null



R19: `municipio_id` deve corresponder a um registro na relação

`ENDERECO_MUNICIPIO`

R20: `bairro_id`  $\neq$  null

R22: `bairro_id` deve corresponder a um registro na relação `ENDERECO_BAIRRO`

R23: `fachada_id` deve corresponder a um registro na relação `AVULSO_ARQUIVO`

R24: `incorporador_id`  $\neq$  null

R25: `incorporador_id` deve corresponder a um registro na relação `INCORPORADOR`

`RESIDENCIAL_ARQUIVO(id, arquivo, create_time, residencial_id, categoria)`

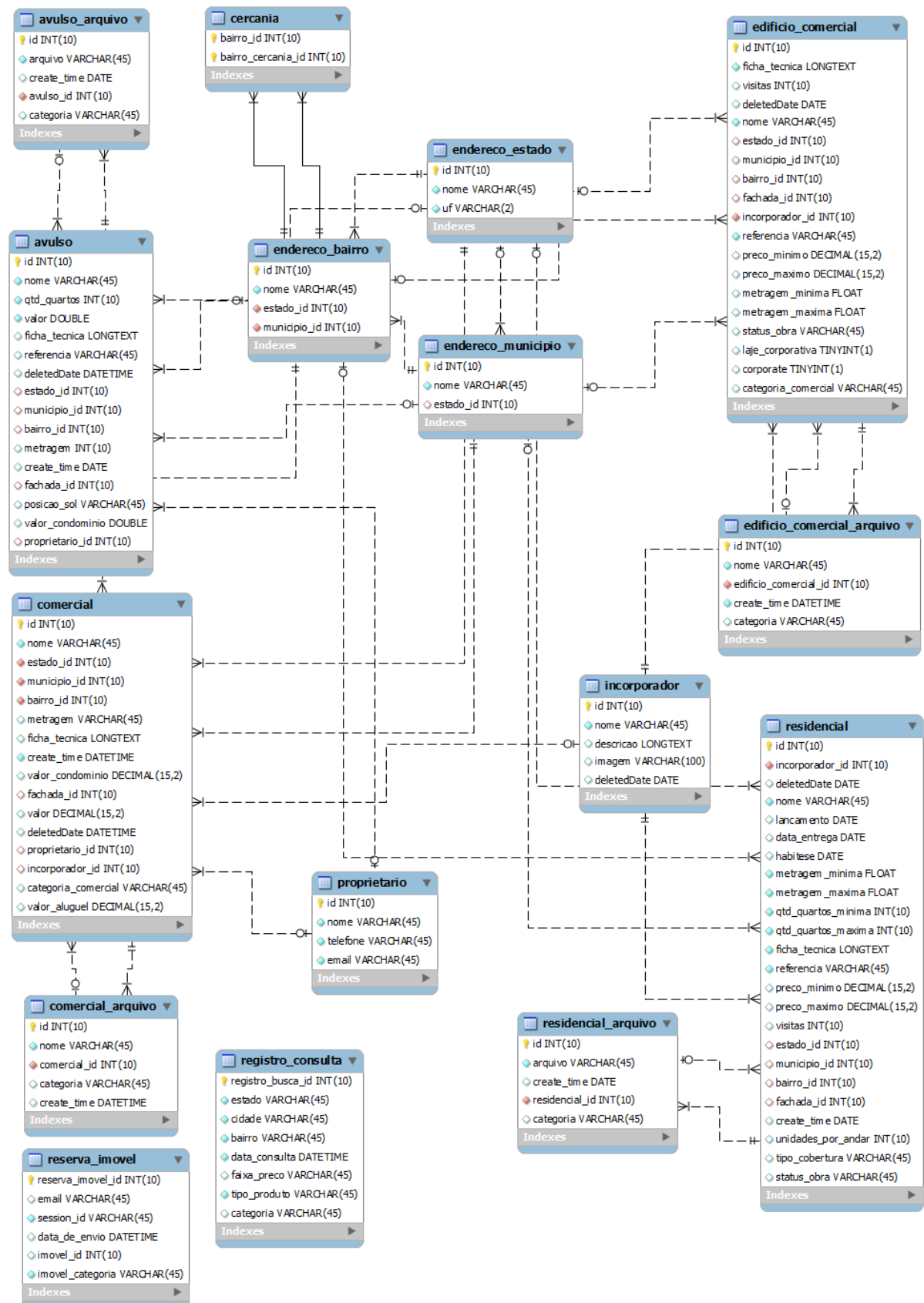
RI1: `create_time` deve ser uma data no formato DATE

RI2: `residencial_id`  $\neq$  null

R3: `residencial_id` deve corresponder a um registro na relação `RESIDENCIAL`

### 3.15 Projeto Físico de Banco de Dados

Projeto físico é a especificação que compreende o modelo lógico de banco de dados garantindo aspectos como performance, padronização, portabilidade e capacidade de recuperação de dados.



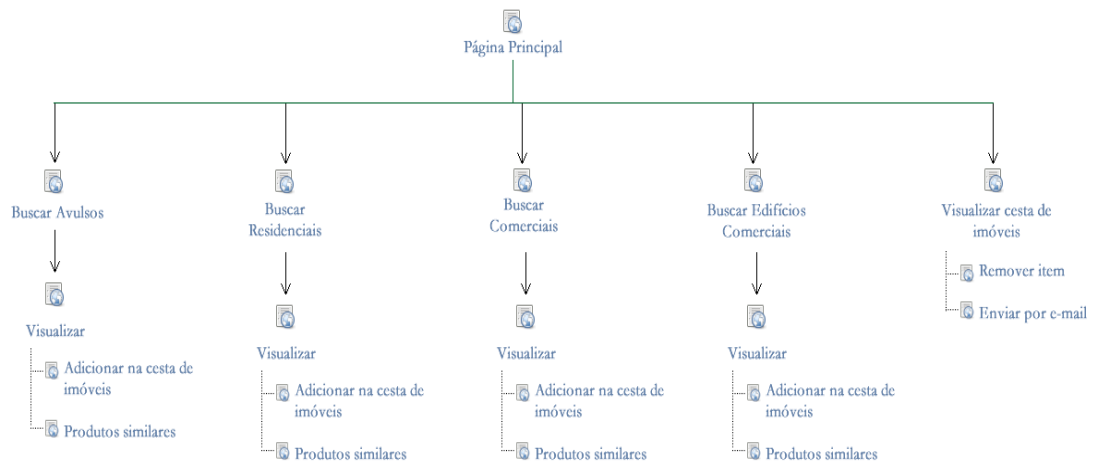
**Figura 27:** Projeto do Banco de Dados

### 3.16 Projeto da Interface Gráfica

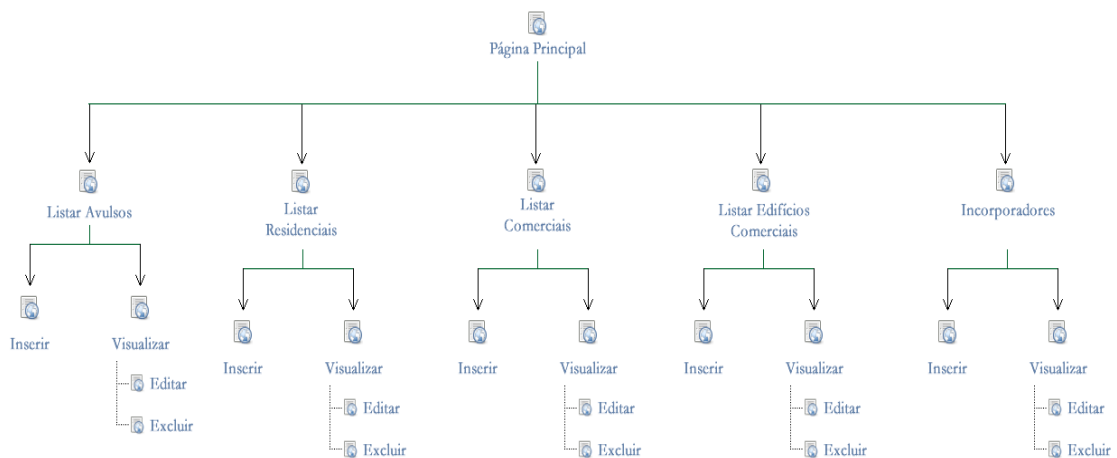
O projeto de interface gráfica aborda os modelos que representam o design em potencial das páginas assim como seus componentes. A interface deve reunir elementos que atendam as necessidades, atitudes e preferências dos usuários do aplicativo conforme eles fazem referência à questão do assunto geral representado pelo site [28].

### 3.17 Hierarquia das telas e mapa de navegação

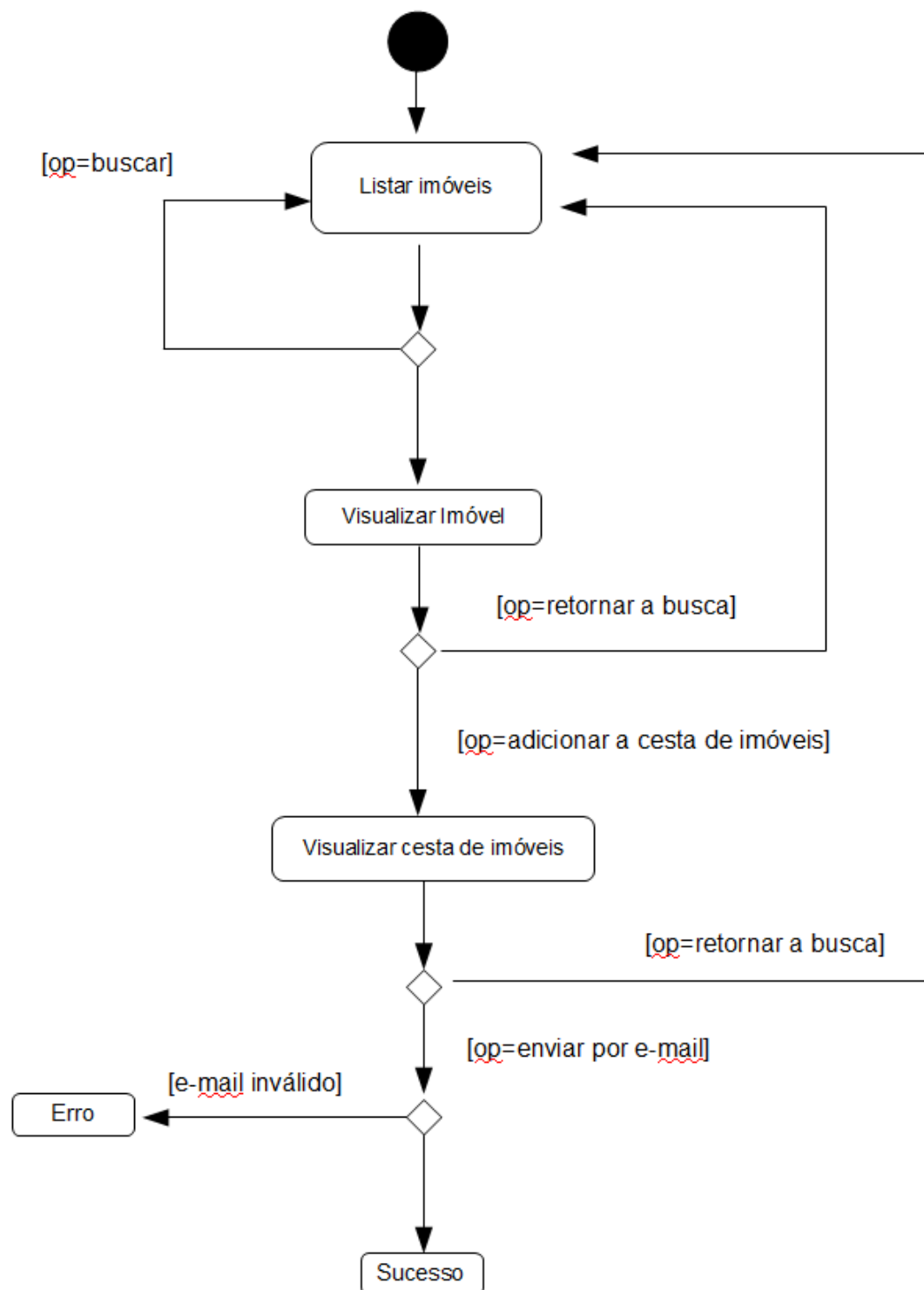
A representação da hierarquia de telas da aplicação, desde a página principal e as outras telas que podem ser acessadas neste compendio estão presentes nesta seção.



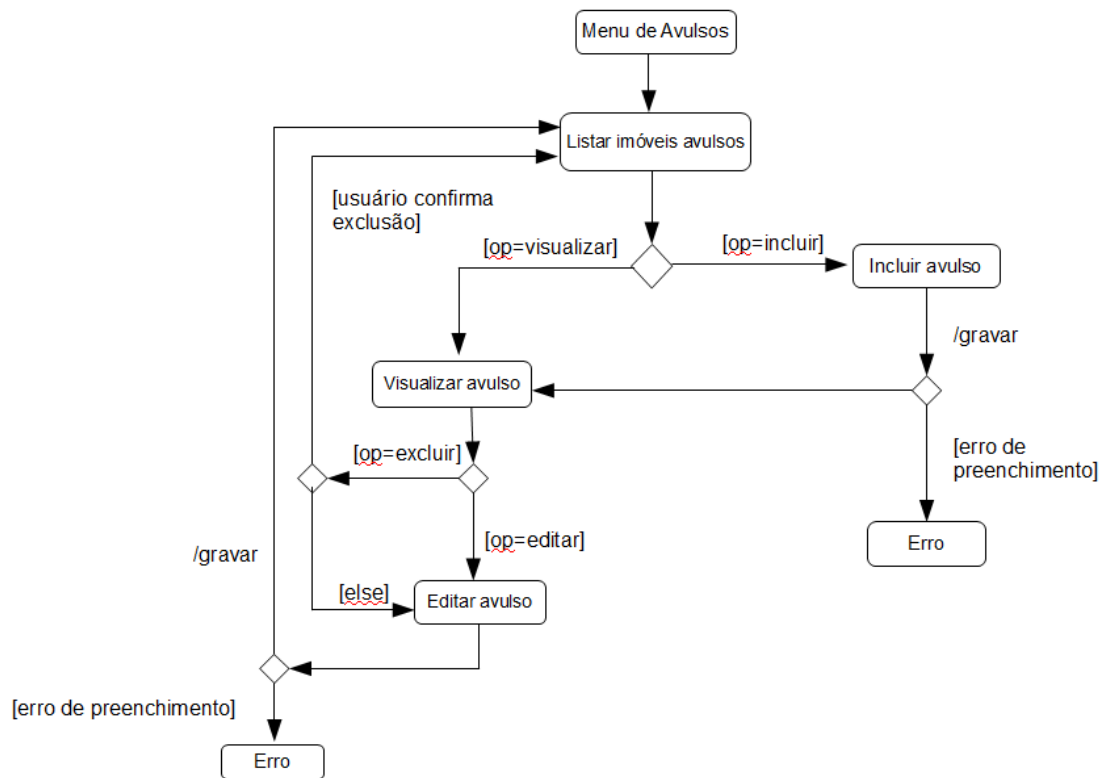
**Figura 28:** Hierarquia das telas na perspectiva do cliente.



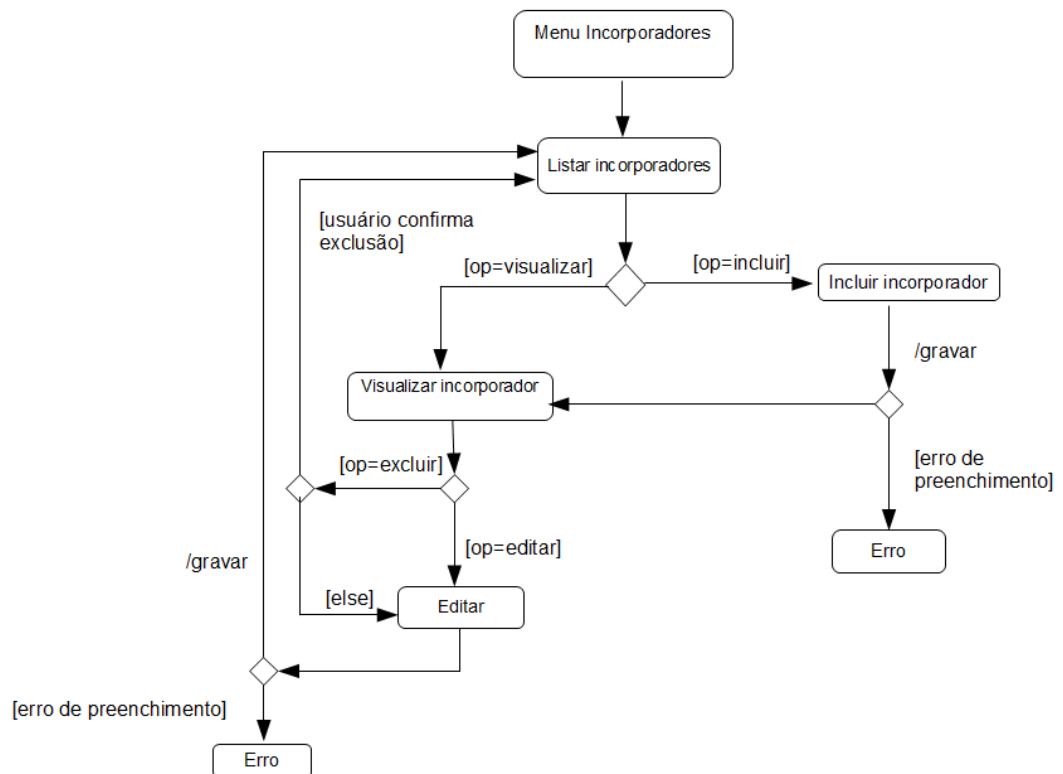
**Figura 29:** Hierarquia das telas na perspectiva do corretor.



**Figura 30:** Mapa de navegação para o caso de uso Manter Cesta de Imóveis.



**Figura 31:** Mapa de navegação para o caso de uso Manter Avulsos.



**Figura 32:** Mapa de navegação para o caso de uso Manter Incorporador.

## Capítulo 4

### Aspectos de Implementação

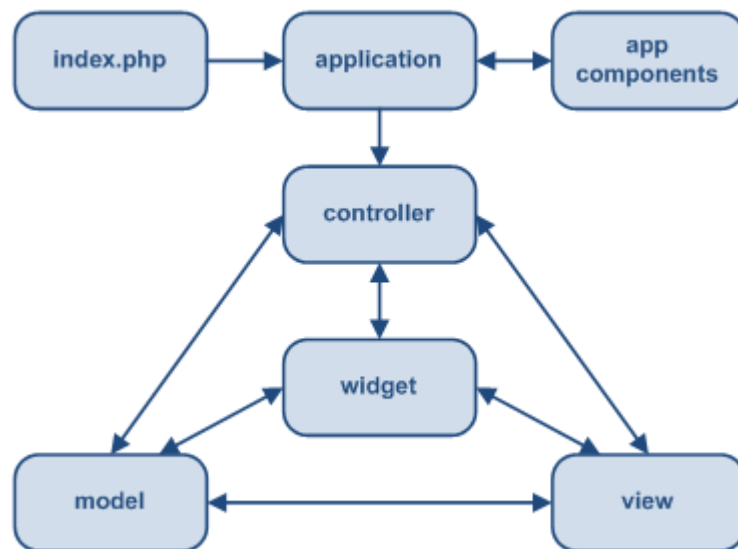
Os aspectos de implementação que foram utilizados durante o processo de desenvolvimento do sistema são apresentados nesta seção. A Seção 4.1 compreende a visão geral da arquitetura da aplicação. A Seção 3.18.2 descreve a estrutura da camada de apresentação da aplicação compreendendo os aspectos específicos deste contexto. A Seção 3.18.3 destaca a estrutura da camada de negócio da aplicação. A Seção 3.18.4 trata da estrutura da camada de persistência da aplicação. Finalmente, a Seção 3.18.5 destaca outros aspectos intrínsecos no desenvolvimento d aplicação.

#### 4.1 Visão geral da arquitetura da aplicação

A implementação de um sistema para internet com a utilização do framework Yii baseia-se na adoção do padrão de projeto (*design pattern*) arquitetural *Model-View-Controller* (MVC), o qual separa a aplicação em três componentes lógicos; *Model*, *View* e *Controller*. O padrão MVC é fortemente adotado em sistemas desenvolvidos para internet, pois consegue separar de maneira simples a lógica de programação da lógica de apresentação.

Além de implementar o padrão MVC, o framework Yii também integra a sua arquitetura o padrão *Front Controller* [29]. Desta maneira o framework concentra a entrada das requisições feitas pelo usuário em um único ponto de acesso. Toda a requisição realizada pelo usuário é capturada por este *Front-Controller*, que age como uma interface genérica no processamento dos comandos.

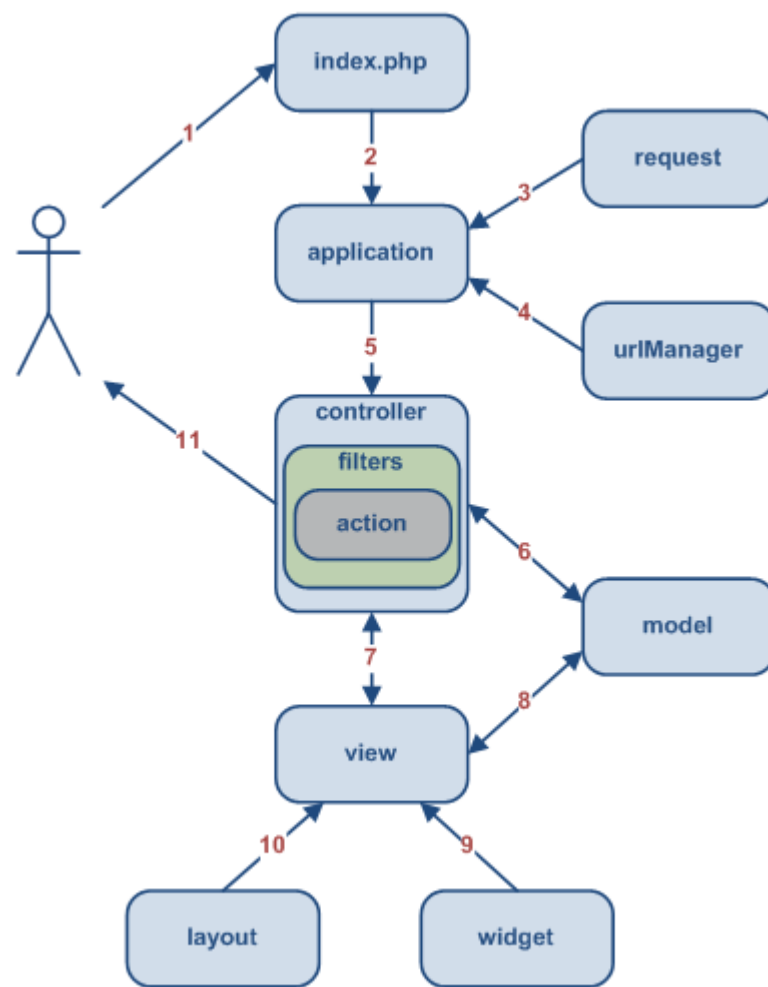
A Figura 33 demonstra a estrutura estática de uma aplicação desenvolvida com o *Framework* Yii:



**Figura 33:** Estrutura de aplicação desenvolvida com framework Yii [20]

A cada nova requisição as informações básicas sobre o pedido são coletadas e despachadas para um controlador adequado, esta execução é realizada pelo script de inicialização (*bootstrap script*) `index.php`. Todo o contexto de execução é encapsulado em um objeto denominado *Application*. Este objeto é criado através da execução do script de inicialização e possui a principal tarefa de despachar as informações de um pedido para um controlador responsável. Este objeto também é responsável por manter as configurações em nível de aplicação e encapsular o acesso a seus componentes. Por esta e outras razões o objeto *Application* também é referenciado como *Front-Controller*.

O script de inicialização garante que a classe *Application* tenha somente uma instância durante a execução da aplicação, fornecendo um ponto global de acesso para ela. Através desta metodologia de inicialização fica clara a utilização do padrão de criação *Singleton* [30] no levantamento do contexto da aplicação. A Figura 34 demonstra como uma requisição é processada por uma aplicação baseada no Framework Yii.



**Figura 34:** Processamento de requisições [20].

A Figura 34 apresenta a execução da requisição em etapas, onde cada uma compreende um processo:

1. Um usuário executa uma requisição com a URL `http://www.example.com/index.php?r=mycontroller/someaction&id=1`, por exemplo, e o servidor web manipula a requisição através da execução do script de entrada;
2. O script de entrada cria uma instância para a classe *Application*;
3. O objeto *Application* obtém as informações relevantes sobre a requisição e as encapsula em um componente denominado *request*;
4. O objeto *Application* encaminha a requisição para o controlador e ação adequados com o auxílio do componente *urlManager*;



5. A aplicação cria a instância do controlador em questão. O controlador por sua vez carrega e executa os filtros de acordo com a ação solicitada. A ação é executada de acordo com os filtros de acesso;
6. A ação carrega um modelo de acordo a id passada pela URL;
7. A ação carrega uma visão para apresentar os dados do modelo;
8. A visão carrega e exibe o modelo com seus dados;
9. A visão pode executar alguns *widgets* caso necessário;
10. A visão é embutida em um layout para apresentação;
11. A ação completa a renderização da visão exibe os dados ao usuário.

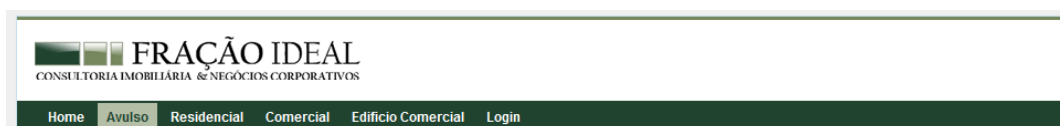
A técnica de persistência e mapeamento de dados adotada ao longo do desenvolvimento da aplicação foi o *Active Record* [29]. Cada classe está relacionada a uma tabela no banco de dados. Cada instância de uma classe *Active Record* representa um registro de uma tabela. Toda instância de uma classe *Active Record* carregada seus atributos de uma determinada tabela. Esta técnica de mapeamento foi adotada pelo framework Yii proporcionando maior produtividade em relação as tarefas mais comuns de inserção e consulta ao banco de dados.

## 4.2 Estruturação da camada de apresentação

O Framework Yii implementa um mecanismo de customização da camada de apresentação estruturado em temas. Muitas idéias e conceitos foram herdados de outros frameworks web bem conhecidos. Em sua arquitetura MVC, a visão corresponde ao componente lógico da camada de apresentação. O Framework Yii integra componentes e *plugins* na visão a fim de que a mesma possa suportar as futuras mudanças de aparência. A camada de apresentação é a interface entre as ações do usuário e o sistema.

O Framework Javascript jQuery foi adotado como a biblioteca oficial para implementação de componentes na camada de apresentação. A biblioteca Javascript oferece componentes e *widgets* reutilizáveis que se integram a qualquer classe *Active Record* da aplicação. Esses componentes correspondem em sua maioria a elementos gráficos de interface, alguns até com implementação de métodos assíncronos.

Toda a requisição executada por um usuário é tratada por um controlador, este por sua vez renderiza um elemento de visão retornando o estado da aplicação após a requisição do usuário. Todo elemento de visão chega ao usuário no formato HTML, e através desses elementos o usuário executa suas primeiras ações. A Figura 35 ilustra o menu da aplicação com a opção Avulso selecionada.



**Figura 35:** Menu da Aplicação.

A Figura 36 ilustra a utilização de componentes da aplicação para a automação da criação de componentes de interface e demonstra a aplicação do princípio DRY (*Don't repeat yourself*), destinado a reduzir a repetição de informações.

```
<div id="mainmenu">
    <?php $this->widget('zii.widgets.CMenu',array(
        'items'=>array(
            array('label'=>'Home', 'url'=>array('/site/index')),
            array('label'=>'Avulso', 'url'=>array('/avulso')),
            array('label'=>'Residencial', 'url'=>array('/residencial')),
            array('label'=>'Comercial', 'url'=>array('/comercial')),
            array('label'=>'Edifício Comercial', 'url'=>array('/edificiocomercial')),
            array('label'=>'Login', 'url'=>array('/site/login'), 'visible'=>Yii::app()->user->isGuest),
            array('label'=>'Logout ('.Yii::app()->user->name.')', 'url'=>array('/site/logout'), 'visible'
        ),
    )); ?>
</div><!-- mainmenu -->
<?php if(isset($this->breadcrumbs)):?>
    <?php $this->widget('zii.widgets.CBreadcrumbs', array(
        'links'=>$this->breadcrumbs,
    )); ?><!-- breadcrumbs -->
<?php endif?>
```

**Figura 36:** Automação dos componentes.

A Figura 37 ilustra o formulário de atualização dos dados de um imóvel Avulso e o menu lateral com a execução das outras atividades do caso de uso Manter Avulso.

**FRAÇÃO IDEAL**  
CONSULTORIA IMOBILIÁRIA & NEGÓCIOS CORPORATIVOS

Home Avulso Residencial Comercial Edifício Comercial Logout (admin)

[Página inicial](#) » [Avulsos](#) » [Meier, 3 quarto\(s\)](#) » Atualizar dados

## Atualizar Meier, 3 quarto(s)

*Campos com \* são obrigatórios.*

Nome \*

Quantidade de quartos \*

Valor \*

Valor do condomínio

Posição do sol

**Operações**

- Listar avulsos
- Inserir novo avulso
- Remover
- Visualizar este avulso

**Figura 37:** Formulário de informações de um imóvel avulso

A Figura 38 apresenta a interface de busca de imóveis residenciais utilizada pelo cliente.

**FRAÇÃO IDEAL**  
CONSULTORIA IMOBILIÁRIA & NEGÓCIOS CORPORATIVOS

Avulsos Residenciais Comerciais Edifícios comerciais

Barra Rio de Janeiro RJ

**Confira os novos produtos residenciais oferecidos pela Fração Ideal**

Exibindo 1-4 de 4 resultado(s).

 Barra Condominium
  Pier da Lagoa
  Blue das Americas

Fração Ideal, a empresa

**Figura 38:** Interface de busca de imóveis.

### 4.3 Estruturação da camada de negócio

Toda a lógica de negócio da aplicação está encapsulada nas classes correspondentes aos modelos dentro da estrutura MVC. Esses modelos por sua vez adotam o padrão *Active Record* encapsulando as operações básicas com o banco de dados através de métodos.

A Figura 39 corresponde ao trecho de código da classe responsável pela realização dos processos de negócio do objeto de domínio Residencial. Este por sua vez é um tipo de imóvel. Herda propriedades e métodos da classe genérica Imovel. Em PHP, classes podem ser extensões de outras classes. As classes estendidas herdam as variáveis e os métodos da classe base. Em PHP toda classe estendida é sempre dependente de uma única classe base. O uso de herança múltipla não é suportado. O relacionamento de herança entre Residencial e Imovel é representado através da palavra-chave *extends*.

```

class Residencial extends Imovel
{
    /**
     * (non-PHPdoc)
     * @see protected/models/Imovel::getEndereco()
     */
    public function getEndereco()
    {
        return "{$this->bairro->nome}, {$this->municipio->nome}, {$this->estado->uf}";
    }

    /**
     * (non-PHPdoc)
     * @see protected/models/Imovel::retrieveAllSimilarData()
     */
    public function retrieveAllSimilarData()
    {
        return KnnResidencial::getSimilar($this);
    }

    /**
     * (non-PHPdoc)
     * @see protected/models/Imovel::getChamadaComercial()
     */
    public function getChamadaComercial()
    {
        return $this->nome;
    }

    /**
     * (non-PHPdoc)
     * @see protected/models/Imovel::getPreco()
     */
    public function getPreco()
    {
        return "a partir de ".NumberDecorator::currency($this->preco_minimo);
    }

    /**
     *
     * Método que dá suporte a consulta ...
     * @param ResidencialConsulta $dados_consulta
     * @return CActiveDataProvider Residencial
     */
    public static function retrieveByExample(ResidencialConsulta $dados_consulta)
    {
        $criteria=new CDbCriteria;

        $criteria->compare('estado_id',$dados_consulta->uf->getPrimaryKey(),true);
        $criteria->compare('municipio_id',$dados_consulta->municipio->getPrimaryKey(),true);
        $criteria->compare('bairro_id',$dados_consulta->bairro->getPrimaryKey(),true);

        if(!is_null($dados_consulta->faixa_preco))
        {
            $preco= explode('-', $dados_consulta->faixa_preco);
            $criteria->addCondition(" {$preco['0']} BETWEEN preco_minimo AND preco_maximo");
            $criteria->addCondition(" {$preco['1']} BETWEEN preco_minimo AND preco_maximo");
        }

        return new CActiveDataProvider('Residencial', array(
            'criteria'=>$criteria,
        ));
    }
}

```

**Figura 39:** Classe Residencial.



clientes estão buscando imóveis com mais frequência, assim como também é possível saber quantos produtos estão sendo oferecidos para em região.

---

```

class TrendReportMonitor extends CActiveRecord
{
    public static function model($className=__CLASS__)
    {
        return parent::model($className);
    }

    public function tableName()
    {
        return 'registro_consulta';
    }

    public static function report(ConsultaForm $form) {

        $data_consulta= date("Y-m-d h:i:s", time());

        $registro= new TrendReportMonitor();
        $registro->setAttribute('estado', $form->uf->uf);
        $registro->setAttribute('cidade', $form->municipio->nome);
        $registro->setAttribute('bairro', $form->bairro->nome);
        $registro->setAttribute('data_consulta', $data_consulta);
        $registro->setAttribute('faixa_preco', $form->faixa_preco);
        $registro->setAttribute('tipo_produto', $form->tipo);
        $registro->setAttribute('categoria', $form->getImovelCategoriaBusca());

        $connection= Yii::app()->db;
        $transaction= $connection->beginTransaction();

        try {
            $registro->save();
            $transaction->commit();
            return true;
        } catch (Exception $e) {
            $transaction->rollback();
        }
    }
}

```

---

**Figura 41:** Classe de Monitoração de Busca.

A Figura 42 apresenta o resultado do processamento dos dados de consulta coletados no site. O relatório é apresentado para o corretor no formato PDF.

## Fração Ideal

### Acompanhamento mensal de tendência das buscas dos consumidores

07/08/2011

RJ/Rio de Janeiro      Méier		Total de consultas	Tendência
		7	alta 100%

Categoria	Faixa de preço	Total de procuras	Produtos correspondentes
residencial	De 100mil até 200mil	5	1
	De 400mil até 500mil	2	1

**Figura 42:** Resultado do processamento dos registros de consulta dos visitantes do site

## 4.4 Estruturação da camada de persistência

A camada de persistência é responsável pelos processos de armazenamento e manutenção do estado dos objetos da aplicação no banco de dados. A utilização do padrão *Active Record* traz para a camada de persistência para o nível do componente lógico Modelo, dentro da arquitetura *model-view-controller* (MVC). Os objetos *Active Record* absorvem a responsabilidade de manter as regras de negócio e persistir seu estado no banco.

A figura 43 apresenta a configuração do componente responsável pela conexão e configuração do banco de dados. As aplicações construídas sobre o framework PHP - Yii concentram os aspectos da aplicação em um só arquivo de configuração. Neste arquivo, cada parâmetro de configuração de um componente é associado a seu *alias*. O *alias* para o componente de conexão ao banco é *db* (referente a classe *CDbConnection*). Sempre que a aplicação for iniciada, o componente *db* será carregado com as configurações de banco representadas na figura 43.



```

40     'components'=>array(
41         'fileManager'=>array(
42             'class'=> 'application.components.SystemService.CFileManager',
43             'upload_path'=>ABSOLUTE_PATH_UPLOAD,
44             'image_crop_configuration'=>array(
45                 array("width"=>145, "height"=>100, "quality"=> 75, "prefix"=>"145_100"),
46                 array("width"=>575, "height"=>458, "quality"=> 75, "prefix"=>"575_458")
47             )
48         ),
49         'user'=>array(
50             'allowAutoLogin'=>true,
51         ),
52         'db'=>array(
53             'connectionString'=>'mysql:host=localhost;dbname=fracao_ideal;',
54             'username'=>'root',
55             'password'=>'root',
56             'charset'=>'latin1'
57         ),
58         'errorHandler'=>array(
59             'errorAction'=>'site/error',
60         ),
61         'log'=>array(
62             'class'=>'CLogRouter',
63             'routes'=>array(
64                 array(
65                     'class'=>'CFileLogRoute',
66                     'levels'=>'error, warning, info',
67                 ),
68             ),
69         ),
70     ),
71 )

```

**Figura 43:** Trecho de código com o mapeamento para o banco de dados MySQL.

A Figura 44 apresenta o código gerado pelo framework para a associação da classe Residencial e sua respectiva tabela no banco de dados.

```

/**
 * Returns the static model of the specified AR class.
 * @return residencial the static model class
 */
public static function model($className=__CLASS__)
{
    return parent::model($className);
}

/**
 * @return string the associated database table name
 */
public function tableName()
{
    return 'residencial';
}

```

**Figura 44:** Código de mapeamento entre uma classe *Active Record* e sua respectiva tabela.

## 4.5 Outros Aspectos

As Subseções a seguir descrevem algumas características e procedimentos relevantes sobre o sistema como *login*, atributos utilizados no processo de recomendação e os procedimentos de implantação da aplicação.

### 4.5.1 Logging

O mapeamento dos processos internos do sistema foi realizado com o componente de Logging do Framework Yii. O componente permite o registro de mensagens nos níveis INFO, WARNING e DEBUG. Também é possível configurar a metodologia de persistência das informações. A Figura 45 apresenta o trecho de código com os parâmetros de configuração do componente de Log da aplicação.

```

40     'components'=>array(
41         'fileManager'=>array(
42             'class'=> 'application.components.SystemService.CFileManager',
43             'upload_path'=>ABSOLUTE_PATH_UPLOAD,
44             'image_crop_configuration'=>array(
45                 array("width"=>145, "height"=>100, "quality"=> 75, "prefix"=>"145_100"),
46                 array("width"=>575, "height"=>458, "quality"=> 75, "prefix"=>"575_458")
47             )
48         ),
49         'user'=>array(
50             'allowAutoLogin'=>true,
51         ),
52         'db'=>array(
53             'connectionString'=>'mysql:host=localhost;dbname=fracao_ideal;',
54             'username'=>'root',
55             'password'=>'root',
56             'charset'=>'latin1'
57         ),
58         'errorHandler'=>array(
59             'errorAction'=>'site/error',
60         ),
61         'log'=>array(
62             'class'=>'CLogRouter',
63             'routes'=>array(
64                 array(
65                     'class'=>'CFileLogRoute',
66                     'levels'=>'error, warning, info',
67                 ),
68             ),
69         ),
70     ),
71 
```

**Figura 45:** Trecho de código com a configuração do Log.

### 4.5.2 Atributos Utilizados no Processo de Recomendação

O processo de recomendação de produtos é executado com base no algoritmo de inteligência artificial *K-NN*. A proximidade (distância) entre dois produtos é calculada a partir da similaridade entre seus conjuntos de atributos numéricos. Cada produto do site atua em conjunto com sua ferramenta de recomendação. Esta por sua vez é especialista nos atributos utilizados no cálculo da distância. Apresentamos os atributos relevantes no cálculo de distância dado seu produto:

- Avulso:
  - Quantidade de quartos;
  - Valor do Imóvel;
  - Metragem;
  - Valor da Cota Condominial;
  - Idade do Imóvel;
  - Valor do IPTU.
- Comercial:
  - Metragem;
  - Valor da Cota Condominial;
  - Valor do Imóvel;
  - Valor do aluguel.
- Edifício Comercial:
  - Faixa de Preço;
  - Metragem das Unidades.
- Residencial:
  - Metragem dos apartamentos;
  - Quantidade de quartos;
  - Faixa de preço das unidades.

Os itens mais similares são pré-selecionados dentre as regiões de áreas contíguas denominadas de cercanias, para então ter sua distância calculada pela implementação do algoritmo K-NN.

### 4.5.3 Procedimentos de Implantação

O ambiente para implantação da aplicação PHP deve obedecer aos seguintes requisitos:

- Servidor Apache;
- PHP 5.2 ou superior;
- Sistema de gerenciamento de banco de dados (SGBD) MySQL;
- O Framework Yii disponível em diretório web acessível;
- Acesso ao servidor via FTP.

A última etapa do processo de instalação consiste em carregar os diretórios do aplicativo no servidor web através de um cliente FTP. É ideal certificar que as diretrizes do arquivo de configuração da aplicação estejam corretas. Após a transferência do aplicativo, o mesmo fica disponível a partir do acesso ao script de entrada `index.php`, por exemplo *<http://<domínio contratado>/index.php>*.

## Conclusão

As conclusões obtidas sobre o desenvolvimento do sistema de recomendação no domínio do mercado imobiliário foram levantadas neste capítulo. A retrospectiva do desenvolvimento do sistema é apresentada na Seção 5.1. Os trabalhos futuros a serem desenvolvidos a partir desta versão do sistema são apresentados na Seção 5.2.

### 5.1 Análise Retrospectiva

Um framework é uma biblioteca que facilita o desenvolvimento de software e reúne padrões de softwares. A abstração das funcionalidades mais comuns no processo de desenvolvimento de software minimizam tarefas repetitivas e auxiliam os programadores a concentrar o desenvolvimento nos processos do negócio. O uso do framework Yii auxiliou nos processos de desenvolvimento técnico, tais como criação da camada de persistência, mapeamento dos dados com os respectivos objetos, reuso de componentes de interface, gerenciamento de temas entre outros. A arquitetura MVC facilitou o tratamento das requisições realizadas pelo usuário assim como a integração da camada de interface com os modelos da aplicação. A utilização do padrão *Active Record* abstraiu os processos mais simples de inserção, alteração, remoção e busca de dados.

### 5.2 Trabalhos Futuros

A aplicação poderá recomendar produtos para um determinado perfil de usuário de maneira automatizada enviando-os por e-mail, como um correio automático, sempre que um novo produto for adicionado.

A aplicação será capaz de manter os proprietários como usuários da aplicação e exibir relatórios com a valorização de seu imóvel de acordo com a procura na região.

## Referências Bibliográficas

- [1] RESNICK, P. ; VARIAN, H. “Recommender systems”, *Commun. ACM*, Vol. 40, N. X, pp. 56-58, MÊS 1997.
- [2] CAZELLA, S.C.; REATEGUI, E.B. “Sistemas de Recomendação”, *In: Congresso da Sociedade Brasileira de Computação XXV Congresso da Sociedade Brasileira de Computação*, São Leopoldo, Vol. 1 pp. 306-348, São Leopoldo, Jul. 2005.
- [3] LINDEN, G. and SMITH, B. and YORK, J. “Amazon.com recommendations: item-to-item collaborative filtering.” *IEEE Internet Computing*, Vol. 7, pp. 76-80, USA, Feb. 2003
- [4] MOONEY, R. J.; ROY, L.; “Content-based book recommending using learning for text categorization.” *SIGIR-99 Workshop on Recommender Systems: Algorithms and Evaluation*, 10.1145 , Berkeley, CA, August 1999.
- [5] MILLER, B. N.; ALBERT, I.; LAM, S. K.; KONSTAN, J. A.;RIEDL, J. ; *MovieLens unplugged: experiences with an occasionally connected recommender system*, University of Minnesota Minneapolis, Miami, Florida, USA : ACM, 2003.
- [6] SCHAFER, B. J.;KONSTAN, J. A.;RIEDL, J.; “E-Commerce Recommendation Applications.” *Data Mining and Knowledge Discover*, Kluwer Academic Publishers Hingham, MA, USA . Janeiro – Abril 2001.
- [7] TORRES, R.; *Personalização na Internet*. 1 ed. Novatec, 2004.
- [8] KING, E. T. ; DAVID;. *Comércio Eletrônico: Estratégia e Gestão*. 1 ed. Prentice-Hall, 2004.
- [9] SVIOKLA; RAYPORT, J.F.; J.J.;. *Exploiting the Virtual Value Chain*. The McKinsey Quarterly. 1996.
- [10] MOREIRA, R.; **Comércio eletrônico está em alta**. Disponível em: <  
[http://www.ebitempresa.com.br/sala\\_imprensa/html/clip.asp?cod\\_noticia=1065&pi=1](http://www.ebitempresa.com.br/sala_imprensa/html/clip.asp?cod_noticia=1065&pi=1)>.  
Acesso em: 05 de ago. 2011.
- [11] RUSSEL, S.; NORVIG, P.; *Inteligência Artificial*. 1 ed. Campus, 2003.
- [12] RABUSKE, R.; *Inteligência Artificial*. Florianópolis, Editora da UFSC, 1995.
- [13] RICH, E.; KNIGHT, K.; *Inteligência Artificial*. São Paulo, Makron Books Editora, 1993.
- [14] SCHAFER, B. J.; KONSTAN, J. A.; RIEDL, J.; “Recommender Systems in E-Commerce”. *ACM CONFERENCE ON E-COMMERCE*. 1999, 2.
- [15] HERLOCKER, L. “Understanding and improving automated collaborative filtering systems.” *Tese - PHD*. Minesota, USA : University of Minnesota, 2000.

- [16] CAZELLA, S.C.; “Aplicando a Relevância da Opinião de Usuários em Sistema de Recomendação para Pesquisadores”. *Tese - Dsc.* Porto Alegre : s.n., 2006.
- [17] RESNICK P., IACOVU N., SUCHAK M., BERGSTROM P. and RIEDL. *GroupLens: An open architecture for collaborative filtering of netnews*. Proceedings of the 1994 Conference on Computer Supported Collaborative Work. 1994.
- [18] GOLDSCHMIDT R., PASSOS E. *Data Mining: Um Guia Pratico*. Rio de Janeiro : Elsevier, 2005.
- [19] THE PHP GROUP. **Manual do Site do PHP Group** In Classes e Objetos. Disponível em <[http://www.php.net/manual/pt\\_BR/language.oop5.php](http://www.php.net/manual/pt_BR/language.oop5.php)>. Acesso em 5 ago.2011.
- [20] YII FRAMEWORK. **Yii Framework Homepage** In The Definitive Guide to Yii. Disponível em <<http://www.yiiframework.com/doc/guide/1.1/en/basics.mvc>>. Acesso em 5 ago.2011.
- [21] ORACLE CORPORATION. **MySQL. The world's most popular open source database.** Site do MySQL. Disponível em: < <http://www.mysql.com/>>. Acesso em: 05 de ago. 2011.
- [22] JQUERY PROJECT. **jQuery: The Write Less, Do More, JavaScript Library.** Site do framework javascript jQuery. Disponível em: <<http://jquery.com/>> Acesso em: 05 de ago. 2011.
- [23]. BOS, B. **Cascading Style Sheets.** World Wide Web Consortium, 2011. Site da World Wide Web Consortium. Disponível em:< <http://www.w3.org/Style/CSS/>>. Acesso em: 05 de ago. 2011 .
- [24] CHANGE VISION. **Change Vision — Astah Community, UML, Professional, Share and iPad.** Disponível em: < <http://astah.net/editions/community> >. Acesso em: 05 de ago. 2011.
- [25] ADOBE SYSTEMS INCORPORATED. **Imagens para tela, prototipagem | Adobe Fireworks CS5.** 2011. Site da Adobe Systems Incorporated. Disponível em:< <http://www.adobe.com/br/products/fireworks.html>>.
- [26]. BEZERRA, E.; *Princípios de Análise e Projetos de Sistemas com UML.* 2 ed.. Rio de Janeiro, Campus, 2007.
- [27] NAVATHE, S.; ELMASRI, R.; *Sistemas de Banco de Dados.* 4 ed. Addison-Wesley, 2005.
- [28] UNGER, R.; CHANDLER, C; *O Guia para Projetar UX.* Alta Books, 2009.
- [29] FOWLER, M; *Patterns of Enterprise Application Architecture.* 1 ed. : Addison-Wesley, 2003.
- [30] VLISSIDES, J.; GAMMA, E.; HELM, R.; JOHNSON;. *Padrões de Projeto - Soluções reutilizáveis de software orientados a objetos.* Alta Books, 2000.

[31] ROSSINI, A.; PALMISANO, A.; *Administração De Sistemas De Informação E A Gestão Do Conhecimento*. 1 ed. Thomson Pioneira, 2003.

[32] LEVINE, R; *Inteligência Artificial e Sistemas Especialistas*. São Paulo, McGraw-Hill, 1988.