

UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO  
ESCOLA DE INFORMÁTICA APLICADA  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Aplicação de Método baseado em Processos de Negócio para Desenvolvimento de  
Serviços

Autores:

Luan Felipe Molnar da Motta Lima

Ricardo Diniz Sul

Orientador:

Leonardo Guerreiro Azevedo

Aplicação de Método baseado em Processos de Negócio para Desenvolvimento de  
Serviços

Luan Felipe Molnar da Motta Lima  
Ricardo Diniz Sul

Projeto de Graduação apresentado à Escola  
de Informática Aplicada da Universidade  
Federal do Estado do Rio de Janeiro  
(UNIRIO) para obtenção do título de  
Bacharel em Sistemas de Informação

Rio de Janeiro - RJ  
Julho/2012

Aplicação de Método baseado em Processos de Negócio para Desenvolvimento de  
Serviços

Aprovado em \_\_\_\_/\_\_\_\_/\_\_\_\_

BANCA EXAMINADORA

---

Prof. Leonardo Guerreiro Azevedo, D.Sc. (UNIRIO)

---

Prof. Flávia Maria Santoro, D.Sc. (UNIRIO)

---

Prof. Fernanda Baião Amorim, D.Sc. (UNIRIO)

Os autores deste Projeto autorizam a ESCOLA DE INFORMÁTICA APLICADA da UNIRIO a divulgá-lo, no todo ou em parte, resguardando os direitos autorais conforme legislação vigente.

Rio de Janeiro, \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

---

Luan Felipe Molnar da Motta Lima

---

Ricardo Diniz Sul

## **AGRADECIMENTOS**

---

Aos nossos pais, José Augusto e Marcia, Ricardo e Francisca, que sempre estiveram ao nosso lado compartilhando tanto dos momentos de alegria como dos momentos difíceis, além de serem a base de nossa vida.

Aos nossos familiares, que foram fundamentais em nossa formação pessoal e profissional e que sempre estiveram conosco nos incentivando.

As nossas namoradas, Marianne e Mariana, pelo carinho, compreensão e paciência.

Ao nosso professor e orientador Leonardo Azevedo, que muito nos ensinou e incentivou, pela confiança e dedicação depositada.

A todo o corpo docente da UNIRIO, pela qualidade do ensino e contribuição para nossa formação.

E, por fim, mas não menos importante, agradecemos a todos os nossos amigos, que dividiram conosco momentos tranquilos e angustiantes, mas que serão sempre lembrados pelos momentos engraçados e de felicidade.

## SUMÁRIO

---

Capítulo 1: Introdução	12
1.1 Motivação	12
1.2 Objetivo	14
1.3 Estrutura do trabalho	14
Capítulo 2: Método para Desenvolvimento de Serviços	16
2.1 Identificação	16
2.2 Análise	17
2.3 Projeto	18
2.4 Implementação	19
Capítulo 3: Enterprise Service Bus	20
3.1 ESBs	20
3.2 Análise de ESBs	21
Capítulo 4: Aplicação do Método	30
4.1 Cenário utilizado na avaliação do trabalho	30
4.2 Identificação	32
4.3 Análise	58
4.4 Projeto	67
4.5 Implementação	68
Capítulo 5: Disponibilização no ESB dos serviços implementados	73
5.1 Executando o servidor de aplicação do ESB	73
5.2 Cadastrando o serviço	73
5.3 Uso do Serviço por Consumidores e Exportação do WSDL	80
Capítulo 6: Aplicação cliente	81
6.1 Implantando a aplicação cliente e serviços no Glassfish	81
6.2 Aplicação cliente dos serviços	83
Capítulo 7: Conclusão	87
Capítulo 8: Referências Bibliográficas	90
Apêndice 1 – Processo Submeter Proposta de Compra	92
Apêndice 2 – Modelos canônicos	104
Apêndice 3 – Modelos físicos	107
Apêndice 4 – Casos de uso e Diagramas de sequência	124
Apêndice 5 – Diagramas de classe	130
Apêndice 6 – Criando um novo domínio no Oracle Service Bus com o assistente de configuração	135
Apêndice 7 – Implementações	142

## LISTA DE FIGURAS

---

Figura 1 - Avaliação dos ESB pela Forrester Research, Inc. (Vollmer <i>et al.</i> , 2011).	28
Figura 2 - Integração entre as diferentes imobiliárias .....	30
Figura 3 - Diagrama do macro processo "Comprar imóvel" .....	31
Figura 4 - Arquitetura utilizada.....	32
Figura 5 - Mapa de granularidade .....	62
Figura 6 - Grupos de entidades .....	64
Figura 7 - Criação das pastas .....	74
Figura 8 - Selecionando o recurso WSDL .....	75
Figura 9 - Definindo os campos do WSDL.....	75
Figura 10 - Selecionando o recurso Proxy Service .....	76
Figura 11 - Definindo os campos do Proxy Service .....	77
Figura 12 - Selecionando o recurso Business Service .....	78
Figura 13 - Definindo os campos do Business Service.....	79
Figura 14 - Definindo o fluxo de mensagens do proxy.....	79
Figura 15 - Definindo o roteamento das mensagens do proxy.....	80
Figura 16 - Exportando o WSDL do serviço .....	80
Figura 17 - Construindo projeto no NetBeans .....	81
Figura 18 - Console do Glassfish .....	82
Figura 19 - Importando .war .....	82
Figura 20 - Página inicial do portal.....	83
Figura 21 - Página de pesquisa do portal .....	84
Figura 22 - Lista de imóveis encontrados .....	84
Figura 23 - Página de detalhes do imóvel .....	85
Figura 24 - Página de submissão de propostas .....	85
Figura 25 - Página com detalhes do cadastro da proposta .....	86
Figura 26 - Processo "Submeter proposta de compra" .....	94
Figura 27 - Pesquisar imóvel.....	95
Figura 28 - Analisar critérios de pesquisa.....	96
Figura 29 - Obter imóveis disponíveis em parceiras.....	97
Figura 30 - Obter imóveis disponíveis .....	98
Figura 31 - Escolher imóvel .....	99
Figura 32 - Submeter proposta de compra .....	100
Figura 33 - Modelo canônico da Imobiliária Online.....	105
Figura 34 - Modelo canônico da Imobiliária Inglesa .....	106
Figura 35 - Modelo canônico da Imobiliária Alemã.....	106
Figura 36 - Modelo físico da Imobiliária Online .....	114
Figura 37 - Modelo físico da Imobiliária Inglesa .....	119
Figura 38 - Modelo físico da Imobiliária Alemã .....	123
Figura 39 - Diagrama de sequência.....	129
Figura 40 - Diagrama de classe do serviço Consolidar Abrangência .....	130
Figura 41 - Diagrama de classe do serviço Consolidar Busca .....	130
Figura 42 - Diagrama de classe do serviço Listar imóveis da imobiliária alemã ....	131
Figura 43 - Diagrama de classe do serviço Listar imóveis da imobiliária inglesa...	131
Figura 44 - Diagrama de classe do serviço Listar imóveis da imobiliária online ....	132
Figura 45 - Diagrama de classe do serviço Registrar critérios de pesquisa .....	132

Figura 46 - Diagrama de classe do serviço Registrar proposta de compra .....	132
Figura 47 - Diagrama de classe do serviço Verificar abrangência da imobiliária alemã .....	133
Figura 48 - Diagrama de classe do serviço Verificar abrangência da imobiliária inglesa .....	133
Figura 49 - Diagrama de classe do serviço Verificar abrangência da imobiliária online.....	134
Figura 50 - Tela inicial do assistente de configuração .....	135
Figura 51 - Selecionando origem de domínio no assistente de configuração .....	136
Figura 52 - Especificando nome e localização do domínio .....	136
Figura 53 - Configurando o nome do usuário e senha do administrador .....	137
Figura 54 - Configurando o modo de inicialização do servidor e JDK .....	138
Figura 55 - Configurando esquema do componente JDBC .....	139
Figura 56 - Teste do esquema do componente .....	139
Figura 57 - Selecionando configurações opcionais.....	140
Figura 58 - Resumo das configurações .....	141
Figura 59 - Serviço Verificar Abrangência da Imobiliária Online .....	142
Figura 60 – DAO do serviço Verificar Abrangência da Imobiliária Online.....	143
Figura 61 - Método Submeter proposta .....	144

## LISTA DE TABELAS

---

Tabela 1 - Análise das ferramentas ESB.....	25
Tabela 2 - <i>Template</i> para identificação dos serviços .....	32
Tabela 3 - Heurística de regra de negócio.....	34
Tabela 4 - Heurística de requisitos de negócio .....	36
Tabela 5 - Heurística de informações de entrada e saída .....	39
Tabela 6 - Heurística de atividades sequenciais.....	41
Tabela 7 - Heurística de OR.....	42
Tabela 8 - Heurística interface de processos.....	43
Tabela 9 - Heurística atividade de múltipla instancia .....	45
Tabela 10 - Eliminação de serviços duplicados .....	46
Tabela 11 - Listagem dos serviços candidatos .....	49
Tabela 12 - Listagem completa dos serviços candidatos .....	49
Tabela 13 - Identificação do grau de reuso para cada serviço candidato.....	53
Tabela 14 - Associação de serviços candidatos com sistemas que os implementam. ....	54
Tabela 15 - Associação de serviços candidatos aos requisitos da demanda .....	55
Tabela 16 - Associação de serviços candidatos com papéis .....	55
Tabela 17 - Associação de serviços às atividades das quais eles foram identificados .....	56
Tabela 18 - Associação de serviços candidatos a partir da origem do serviço .....	57
Tabela 19 - Agrupamento por grau de reuso de serviços candidatos.....	58
Tabela 20 - Agrupamento por sistemas que implementam serviços candidatos.....	59
Tabela 21 - Serviços candidatos identificados a partir de atividades de múltiplas instâncias .....	59
Tabela 22 - Agrupamento de serviços candidatos associados aos requisitos da demanda .....	60
Tabela 23 - Agrupamento de papéis que utilizam (executam) o serviço .....	60
Tabela 24 - Cálculo de priorização de serviços .....	61
Tabela 25 - Priorização de serviços identificados a partir de fluxo .....	62
Tabela 26 - Associação de serviços candidatos com grupos de entidades.....	65
Tabela 27 - Agrupamento de serviços candidatos de acordo com grupos de entidades manipuladas.....	65
Tabela 28 - Seleção de serviços .....	66
Tabela 29 - Identificação dos tipos de serviços .....	67
Tabela 30 - Serviços implementados .....	70
Tabela 31 - Notação dos elementos utilizados na modelagem de processos .....	92
Tabela 32 - Descrição do Requisito de Negócio "Registrar critérios de pesquisa por imóveis" .....	95
Tabela 33 - Descrição do Requisito de Negócio "Verificar abrangência de imobiliárias parceiras" .....	96
Tabela 34 - Descrição da Regra de Negócio "Abrangência de imobiliária parceira" .....	96
Tabela 35 - Descrição do Requisito de Negócio "Consultar imóveis" .....	97
Tabela 36 - Descrição do Requisito de Negócio "Consultar imóveis disponíveis" ...	98
Tabela 37 - Descrição do Requisito de Negócio "Listar imóveis" e "Reservar imóvel" .....	99
Tabela 38 - Descrição do Requisito de Negócio "Registrar proposta de compra" ..	100



Tabela 39 - Descrição dos clusters do processo "Submeter proposta de compra"... 100

## LISTA DE ABREVIATURAS

---

BAM - *Business Activity Monitoring*  
BPEL - *Business Process Execution Language*  
COTS - *Commercial Off-The-Shelf*  
CRUD – *Create, Retrieve, Update and Delete*  
CSV - *Comma Separated Values*  
ESB – *Enterprise Service Bus*  
FTP - *File Transfer Protocol*  
HTTP/S - *Hypertext Transfer Protocol / Secure*  
IMAP - *Internet Message Access Protocol*  
JBI - *Java Business Integration*  
JDBC - *Java Database Connectivity*  
JMS - *Java Message Service*  
JMX - *Java Management eXtensions*  
JSON - *JavaScript Object Notation*  
OSGi - *Open Services Gateway Initiative*  
POJO - *Plain-Old Java Object*  
POP3 - *Post Office Protocol*  
REST - *Representational State Transfer*  
SOA - *Service Oriented Architecture*  
SOAP - *Simple Object Access Protocol*  
SMTP - *Simple Mail Transfer Protocol*  
TCP - *Transmission Control Protocol*  
TI – *Tecnologia da Informação*  
WS-BPEL - *Web Services Business Process Execution Language*  
WSDL - *Web Service Definition Language*  
XML - *eXtensible Markup Language*  
XSLT - *eXtensible Stylesheet Language for Transformation*

## RESUMO

---

O conceito de SOA tem se mostrado fundamental para as organizações que desejam uma melhor adaptação às constantes mudanças no meio onde atuam. Serviços muitas vezes são desenvolvidos de forma incorreta e inconsistente. Este trabalho apresenta a aplicação prática dos métodos para desenvolvimento de serviços propostos por Azevedo *et al.* (2009a, 2009b, 2011 e 2012) e a utilização de um ESB para apoiar a implantação destes serviços. O ESB utilizado neste trabalho foi o Oracle Service Bus, presente no pacote Oracle SOA Suite, escolhido após análise entre ferramentas comerciais e de código aberto.

**Palavras-chave:** SOA, ESB, Método para desenvolvimento de serviços, Oracle Service Bus, Oracle SOA Suite.

# CAPÍTULO 1: INTRODUÇÃO

---

O objetivo deste capítulo é contextualizar o assunto abordado neste trabalho, bem como apresentar sua motivação, objetivo e a estrutura dos capítulos.

## 1.1 Motivação

O ambiente organizacional se altera constantemente. Devido a estas mudanças, surgem oportunidades e chances para estas organizações, assim como riscos e ameaças. De forma a lidar com estas mudanças e aumentar a agilidade e a capacidade de resposta das organizações, surge a ideia da arquitetura orientada a serviços (SOA - *Service-Oriented Architecture*) (Kohlborn *et al.*, 2009).

Na literatura existem diversas definições sobre SOA, além do mais na área de Sistemas de Informação cada projeto possui características únicas, o que implica em aplicações SOA com diferentes resultados de acordo com o ponto de vista de cada pesquisador (Erickson e Siau, 2008).

Segundo Josuttis (2007), SOA é um paradigma para a realização e a manutenção dos processos corporativos que se encontram em grandes sistemas distribuídos.

SOA pode utilizar como insumo processos de negócio e esses processos são executados em diferentes etapas (atividades ou tarefas) e em diferentes sistemas. Um serviço representa uma funcionalidade do negócio, ou seja, deve representar uma funcionalidade autocontida que corresponda a uma atividade do mundo real (Josuttis, 2007). Segundo Erl (2005), serviços correspondem a recursos de software bem definidos através de uma linguagem padrão e independem do estado ou contexto de outros serviços. Marks e Bell (2006) apresentam que serviços correspondem a pedaços de funcionalidades que têm suas interfaces expostas e se comunicam via mensagens.

Organizações possuem inúmeros processos, a partir dos quais se pode desenvolver serviços. Com isto, muitas vezes, mais de um serviço pode implementar a mesma funcionalidade se uma análise adequada não for realizada ou mesmo funcionalidades podem ser implementadas como serviços quando na realidade seria

mais adequado disponibilizá-las como funcionalidades de aplicações tradicionais. Neste último caso, a funcionalidade será disponibilizada como serviço, mas não será utilizada em sua plenitude para alcançar os objetivos de SOA, como, por exemplo, não ter reuso por outras aplicações ou corresponder a uma funcionalidade não durável. Entende-se como durável uma funcionalidade que persiste na organização ao longo do tempo. Como exemplo, em uma seguradora, o cálculo do valor do sinistro sempre irá existir, embora a forma de cálculo seja alterada ao longo do tempo.

Desta forma, identifica-se a necessidade de se ter um método para o desenvolvimento desses serviços, evitando-se problemas como, por exemplo, serviços duplicados, sem reuso, sem amplo uso ou com uso por um período muito curto de tempo. Uma das formas é basear o método de desenvolvimento de serviços em modelos de processo de negócio, visto que este último contém diversas informações detalhadas como, por exemplo, regras de negócio, requisitos de negócio, atividades, informações de entrada e saída, entidades do negócio, fluxo de atividades, as quais são muito úteis nas etapas de identificação e análise dos serviços auxiliando no alinhamento de TI com negócio, como preconizado por Gu e Lago (2007). Existem diferentes métodos para desenvolvimento de serviço tendo como base informações presentes em modelos de processos de negócio. Azevedo *et al.* (2009a, 2009b) apresentam uma análise da literatura em relação a métodos para identificação de serviços e concluem que falta sistemática nos métodos ou apresentam apenas diretrizes ou o nível de detalhes não é suficiente para aplicá-los na prática. Os mesmos problemas foram identificados por Azevedo *et al.* (2011) e Diir *et al.* (2012) em relação aos métodos de análise, projeto e implementação de serviços.

Neste trabalho, os métodos de Azevedo *et al.* (2009a, 2009b e 2011) e Diir *et al.* (2012) foram escolhidos para serem empregados. Esta escolha se deve ao fato destes métodos apresentarem heurísticas sistemáticas que podem ser aplicadas na prática. Além disso, estes trabalhos estão contemplados em projetos de pesquisa dos professores do Departamento de Informática Aplicada (DIA) da UNIRIO, dessa forma, servindo como continuidade e validação de aspectos importantes destes trabalhos.

Este trabalho executa todas as heurísticas dos métodos, mas tem foco principal na implementação e disponibilização dos serviços. Além disso, neste trabalho a disponibilização dos serviços será feita com a utilização de um *Enterprise Service Bus* (ESB) o que não foi abordado nas referências utilizadas como base para este trabalho. ESB é considerado a principal tecnologia para a implantação de uma abordagem SOA (Hewitt, 2009). Segundo Papazoglou *et al.* (2007), o ESB é um barramento de mensagens baseado em padrões que permitem a implementação, disponibilização e gerenciamento de soluções SOA, sendo projetado para prover a interoperabilidade entre aplicações utilizando adaptadores e interfaces.

## 1.2 Objetivo

Este trabalho tem o objetivo de avaliar os métodos para desenvolvimento de serviços propostos por Azevedo *et al.* (2009a, 2009b, 2011) e Diirr *et al.* (2012) através da execução prática dos passos propostos. Busca-se identificar pontos fortes e fracos de cada proposta a saber: (i) Identificação de serviços, (ii) Análise de serviços, (iii) Projeto de serviços e (iv) Implementação de serviços.

Além disso, este trabalho apresenta a evolução da proposta considerando o uso de ESB, uma vez que os próprios autores apontaram isto como um dos trabalhos futuros em Diirr *et al.* (2012).

Um processo de negócio que representa um cenário fictício de venda de imóveis é utilizado neste trabalho para exemplificar a aplicação das heurísticas bem como auxiliar na análise dos métodos propostos.

## 1.3 Estrutura do trabalho

Este trabalho está dividido da seguinte forma. O capítulo 1 corresponde a presente introdução, apresentando a motivação, o objetivo e contextualizando os assuntos abordados ao longo do trabalho. O capítulo 2 apresenta o método para desenvolvimento de serviços aplicado no trabalho. No capítulo 3 são apresentados os conceitos de ESB, além de mostrar uma análise de ferramentas comerciais e de código aberto disponíveis. O objetivo desta análise foi identificar a mais adequada para a execução deste trabalho. O capítulo 4 apresenta a aplicação do método para

desenvolvimento de serviços. O capítulo 5 apresenta a implementação dos serviços no ESB. No capítulo 6 é apresentada a aplicação cliente, responsável por consumir os serviços implementados no ESB. Por fim, o capítulo 7 apresenta a conclusão do trabalho e o capítulo 8 as referências bibliográficas.

Os códigos-fonte da implementação desenvolvida neste trabalho encontram-se em um repositório online e podem ser acessados no endereço *<http://code.google.com/p/pf-aplicacao-metodo-bpm-soa/>*.

## CAPÍTULO 2: MÉTODO PARA DESENVOLVIMENTO DE SERVIÇOS

---

Neste capítulo apresentaremos de forma resumida as heurísticas dos métodos empregados neste trabalho. Maiores detalhes são apresentados quando da aplicação prática do método.

Este trabalho tem como foco os métodos propostos por Azevedo *et al.* (2009a, 2009b e 2011), o qual engloba as etapas de identificação e análise dos serviços, e no método proposto por Diirr *et al.* (2012), que por sua vez descreve as etapas de projeto e implementação dos serviços, abrangendo desta forma etapa de suma importância para o desenvolvimento de serviços.

Vale ressaltar que o método proposto por Azevedo *et al.* (2009a, 2009b e 2011) possui uma abordagem *top-down*, ou seja, inicia-se com uma visão de alto nível e acrescentam-se detalhes ao longo do método. Neste caso, a entrada utilizada são modelos de processos de negócio.

As etapas dos métodos citados são brevemente detalhadas a seguir.

### 2.1 Identificação

A identificação de serviços candidatos (Azevedo *et al.* 2009a, 2009b) possui como entrada modelos de processos de negócio e gera como saída um conjunto de serviços candidatos com informações consolidadas dos mesmos a fim de auxiliar o analista SOA a definir a melhor implementação para as funcionalidades do negócio, e pode ser dividida em três passos: selecionar as atividades; identificar e classificar os serviços candidatos; e consolidar os serviços candidatos.

Segundo Erl (2005), um serviço candidato é uma abstração de um serviço que, durante a fase de projeto, pode ser escolhido para ser implementado como serviço ou função de uma aplicação.

O primeiro passo se resume em selecionar as atividades a partir dos modelos. As atividades selecionadas são aquelas que são totalmente executadas por sistemas ou executadas por pessoas com auxílio de sistemas ou candidatas para automação.



No segundo passo, deve-se utilizar a lista de atividades obtidas no passo anterior para identificar serviços com o uso de heurísticas que analisam elementos semânticos das atividades (tais como, regra de negócio, requisitos de negócio e informações de entrada e saída) e as estruturas de fluxos do processo (tais como atividades sequenciais, *OR*, *AND*, *XOR*) para identificar e classificar os serviços como serviços candidatos de dados (realizam operações CRUD – *Create*, *Retrieve*, *Update* e *Delete* sobre dados) ou de lógica (encapsulam regras de negócio, podendo realizar acesso ao dado).

O terceiro passo realiza a consolidação destes serviços, através da aplicação de heurísticas a fim de produzir informações disponibilizadas em grafos, gráficos e tabelas para auxiliar o analista SOA a decidir quais serviços candidatos devem ser implementados como serviços físicos ou como funcionalidades de aplicações. Exemplos de heurísticas são: eliminar serviços duplicados; definir o grau de reuso dos serviços; associar os serviços com atividades, sistemas e fluxos que o executem.

Ao final deste passo, tem-se uma lista de serviços candidatos e informações sobre os mesmos, que são utilizadas nos passos seguintes.

## **2.2 Análise**

A análise de serviços candidatos (Azevedo *et al.*, 2011) é dividida em três passos: priorizar serviços candidatos; definir a granularidade dos serviços candidatos; e agrupar os serviços candidatos.

O primeiro passo tem o objetivo de definir a prioridade dos serviços candidatos. Isto é feito a partir da soma dos valores (pesos) estabelecidos para cada informação relevante do serviço, tais como: grau de reuso; associação com sistemas; requisições de demanda; função organizacional; e se o serviço foi identificado a partir de atividades de múltiplas instâncias.

No segundo passo, mapas de granularidade são elaborados a fim de separar os serviços de granularidade fina e de granularidade grossa, além de explicitar dependências entre os serviços.

O terceiro passo consiste em agrupar os serviços candidatos de acordo com as entidades que estes manipulam. Neste passo, os mapas de granularidade são

utilizados para auxiliar no agrupamento de serviços candidatos de dados e serviços candidatos de lógica.

Ao final, têm-se grupos de serviços candidatos, estando estes grupos de acordo com as normas da organização.

## **2.3 Projeto**

O projeto de serviços é dividido em quatro passos: separar serviços de dados de serviços de lógica para implementação; elaborar o modelo canônico contendo as entidades manipuladas pelos serviços; definir as operações dos serviços que irão automatizá-los; e modelar os serviços.

O primeiro passo deve ser realizado dado que serviços de dados e serviços de lógica possuem diferentes características e, portanto, devem ser implementados de formas diferentes.

No segundo passo, o modelo canônico deve ser elaborado, a fim de evitar transformação de dados na comunicação entre serviços. Gilpin [2010] define o modelo canônico como um modelo da semântica e estrutura de informações da organização. Este modelo está aderente a um conjunto de regras em um contexto definido para comunicação entre um conjunto de aplicações ou partes.

O terceiro passo consiste na definição de como será a implementação dos serviços, que foram selecionados no passo de análise de serviços para serem implementados em serviços físicos. As possibilidades são as seguintes: um serviço candidato pode ser implementado como uma operação de serviço físico; mais de um serviço candidato pode ser implementado em uma única operação de serviços físico; ou um serviço candidato pode ser decomposto em mais de uma operação de serviço físico.

O quarto passo realiza a modelagem dos serviços utilizando UML. Neste passo devem ser elaborados: diagrama de classes para detalhar serviços e entidades em classes e explicitar dependências entre serviços; diagramas de atividades, a fim de se obter uma visão do fluxo de operações; além de casos de uso para detalhar operações e fluxo de operações a fim de facilitar a implementação dos serviços.

Ao final desta etapa espera-se ter uma visão mais detalhada sobre os serviços e suas operações para seguir para a implementação.

## 2.4 Implementação

A etapa de implementação é dividida em dois passos: implementar entidades do negócio; e implementar serviços em si.

No primeiro passo, devem ser implementadas as classes POJO (*Plain-Old Java Object*) para as entidades presentes no modelo canônico. Estas entidades representam os tipos de dados manipulados pelo negócio e, em geral, estarão ligadas diretamente aos dados armazenados no banco de dados. Em outras palavras, dados do banco de dados serão lidos para objetos destas entidades para serem manipulados pela aplicação, assim como os dados serão escritos no banco de dados a partir das mesmas.

No segundo passo, devem ser implementados os serviços utilizando tecnologia para implementação de serviços como, por exemplo, a tecnologia de *Web services*. Erl (2005) aponta que esta tecnologia é a principal tecnologia para implementação de serviços.

Ao final desta etapa, tem-se os serviços implementados.

## CAPÍTULO 3: ENTERPRISE SERVICE BUS

---

Este capítulo apresenta a definição de ESB e uma análise de um conjunto de ESBs comerciais e de código aberto para avaliar o mais adequado para a publicação dos serviços implementados neste trabalho.

### 3.1 ESBs

Hewitt (2009) defende que o ESB (*Enterprise Service Bus*) é a principal tecnologia em uma abordagem SOA. O ESB tem como objetivo fornecer uma camada de comunicação e interpretação lógica entre consumidor e provedor de serviços, permitindo assim que clientes não possuam uma referência direta à localização do serviço. Isto diminui o acoplamento, pois evita dependências diretas consumidor/serviço.

Segundo Josuttis (2007), a partir das abordagens técnicas e organizacionais utilizadas para implementar o ESB, ele tem como responsabilidade prover as seguintes funcionalidades:

- Prover conectividade: prover a comunicação entre aplicações desenvolvidas em diferentes linguagens e plataformas;
- Transformar dados: como o ESB promove a conectividade entre diferentes plataformas, é responsabilidade do ESB transformar mensagens enviadas por um consumidor do serviço para o formato esperado pelo serviço bem como realizar as transformações necessárias no envio da mensagem de resposta do serviço para o consumidor;
- Roteamento: o ESB tem como tarefa o roteamento inteligente como, por exemplo: balanceamento de carga, roteamento baseado em conteúdo e tratamento de falhas;
- Tratar segurança: prover a comunicação segura entre as diversas aplicações;

- Tratar confiabilidade: prover a garantia da confiabilidade nas mensagens trafegadas entre consumidores e serviços;
- Gerenciar serviços: prover a publicação dos diversos serviços e suas respectivas versões;
- Monitorar serviços: tem como objetivo acompanhar a execução dos serviços fornecendo alertas caso algum evento aconteça, monitorar o fluxo de mensagens, a carga de dados que é trafegada no ESB, etc.

## 3.2 Análise de ESBs

Existem inúmeros ESBs disponíveis no mercado tanto comerciais como de código aberto. A análise deste projeto se focou na popularidade e facilidade de uso dos ESBs, observando-se alguns fatores como: a disponibilização de tutoriais e documentação; a frequência de atualizações; a relevância da ferramenta no mercado; entre outros. Desta forma, buscou-se escolher a ferramenta que melhor se adequasse à disponibilização dos serviços implementados neste trabalho, provendo as responsabilidades de ESB descritas na seção 3.1.

Os ESBs comerciais analisados foram:

Microsoft Biztalk<sup>1</sup> – Microsoft Biztalk é uma ferramenta ESB que permite a integração e conectividade de servidores, possuindo adaptadores para diversas outras plataformas, além de ter uma infraestrutura de mensagens robusta. Dentre suas funcionalidades, o Biztalk fornece um motor de regras, BAM (*Business Activity Monitoring*), entre outras. Esta ferramenta é compatível apenas com o Windows.

Oracle Service Bus<sup>2</sup> – Oracle Service Bus é uma extensão do pacote Oracle SOA Suite, aumentando sua performance e escalabilidade. Com esta ferramenta é possível integrar redes realizando a conexão, mediação e gerenciamento das interações entre os serviços e as aplicações. Esta ferramenta está disponível em versões para Windows e Linux, além de uma versão Genérica.

Os ESBs de código aberto analisados foram:

---

<sup>1</sup> <http://www.microsoft.com/biztalk/en/us/default.aspx>

<sup>2</sup> <http://www.oracle.com/technetwork/middleware/service-bus/overview/index.html>

Apache ServiceMix<sup>3</sup> - Apache ServiceMix é um módulo de integração de código aberto que unifica os recursos e funcionalidades de outras ferramentas da Apache em uma plataforma de execução que permite ser usada para construir soluções de integração. Ele está sob a licença Apache v2 e suas principais características são: troca de mensagens, roteamento e implementação de padrões de integração empresarial; suporta disponibilização de serviços REST (*Representational State Transfer*); permite integração flexível entre outros componentes; completa à engine WS-BPEL (*Web Services Business Process Execution Language*) do Apache ODE; baseado no Apache Karaf OSGi (*Open Services Gateway initiative*) para implantação de aplicações; Compatível com Windows, Linux e Mac.

Apache Synapse Enterprise Service Bus<sup>4</sup> - Apache Synapse é um barramento ESB alimentado por um motor assíncrono que oferece suporte para XML (*eXtensible Markup Language*), Web Services e REST. Além de XML e SOAP (*Simple Object Access Protocol*), o Apache Synapse suporta outros formatos de conteúdo para integração, tais como texto simples, binário, Hessian e JSON (*JavaScript Object Notation*). Há diversos adaptadores para o Apache Synapse, permitindo a comunicação do mesmo com muitas aplicações e protocolos da camada de transporte, como, por exemplo, suporte a HTTP/S (*Hypertext Transfer Protocol / Secure*), e-mail (POP3, IMAP, SMTP), JMS (*Java Message Service*). Apache Synapse é um software de código aberto distribuído sob a licença da Apache Software License 2.0. Possui versões disponíveis para Windows e Linux.

Mule ESB<sup>5</sup> - Mule ESB é um ESB baseado em Java que permite a integração com sistemas existentes, independente de tecnologia, incluindo JMS, Web Services, JDBC (*Java Database Connectivity*), HTTP, e outros. Dentre as funcionalidades do Mule ESB, podemos destacar os seguintes recursos: criação e hospedagem de serviços; mediação de serviços; roteamento de mensagens; transformação de dados; compatibilidade com Windows, Linux e Mac; interface gráfica de instalação e para uso do ESB, similar à de um *plug-in* do eclipse.

---

<sup>3</sup> <http://servicemix.apache.org/>

<sup>4</sup> <http://synapse.apache.org/>

<sup>5</sup> <http://www.mulesoft.org/what-mule-esb>

FuseSource<sup>6</sup> - FuseSource é uma plataforma de integração de código aberto baseada no Apache Service MIX. O FuseSource tem uma arquitetura integrável que permite que as organizações usem suas soluções de serviços principais em SOA. O motor do serviço ou componente de interconexão é compatível com padrões JBI (*Java Business Integration*) e OSGi - incluindo BPEL (*Business Process Execution Language*), XSLT (*eXtensible Stylesheet Language for Transformation*) ou motores JMX (*Java Management Extensions*) - que podem ser implantados em um repositório FuseSource. Além disso, componentes FuseSource podem ser implantados em outros ESBs. Este ESB possui versões disponíveis para Windows, Linux e Mac.

Petals ESB<sup>7</sup> - Permite o desenvolvimento de projetos distribuídos em diversos servidores e é compatível com os padrões da indústria (como JBI, SCA, BPEL ou WSDL), além de possuir suporte para a adaptação em grandes infraestruturas e implementação da arquitetura em ferramentas de código aberto. Este ESB possui versões disponíveis para Windows e Linux.

WSO2 Enterprise Service Bus<sup>8</sup> - WSO2 Enterprise Service Bus é um ESB que permite a exposição de aplicações existentes e serviços através de diferentes protocolos e formatos, permite o balanceamento de carga, permite o gerenciamento de segurança de forma centralizada, criação de *logs*, monitoramento e outros. Dentre suas funcionalidades podemos destacar: conexão com protocolos da camada de transporte como HTTP, HTTPS, POP, IMAP, etc.; conectividade com os formatos e protocolos JSON, XML, SOAP 1.1, SOAP 1.2, etc.; possui adaptadores para sistemas COTS (*Commercial off-the-shelf*), como MS Navision, IBM WebSphere MQ, Oracle AQ, etc.; possui adaptadores para serviços na nuvem; roteamento, mediação e transformação; compatível com Windows e Linux.

UltraESB<sup>9</sup> - O UltraESB é um ESB de código aberto que integra diferentes sistemas e tem suporte à validação de mensagens recebidas, verificações de segurança e outros. Entre suas funcionalidades possui suporte a: HTTP/S, JMS, E-mail, TCP (*Transmission Control Protocol*), MLLP/S, FTP/S (*File Transfer*

---

<sup>6</sup> <http://fusesource.com/products/enterprise-servicemix/>

<sup>7</sup> <http://petals.ow2.org/>

<sup>8</sup> <http://wso2.com/products/enterprise-service-bus/>

<sup>9</sup> <http://adroitlogic.org/>

*Protocol*), SFTP, SOAP, REST, XML, HTML, JSON, CSV (*Comma Separated Values*), entre outros, além de possuir versões compatíveis com Windows e Linux.

De forma a facilitar a comparação entre os ESBs, a Tabela 1 apresenta uma análise de algumas das características importantes na escolha de um ESB: o status da documentação; últimas atualizações; nível de atividade no fórum; apresentação de exemplos e/ou tutoriais de cada ESB analisado.



**Tabela 1 - Análise das ferramentas ESB**

	<b>Documentação</b>	<b>Atualização</b>	<b>Fórum</b>	<b>Exemplos/ Tutoriais</b>	<b>Sistemas operacionais</b>
<b>Microsoft Biztalk</b>	Atualizada	Última atualização em Setembro/2010	Em atividade	Possui	Windows
<b>Oracle Service Bus</b>	Atualizada	Última atualização em Janeiro/2012	Em atividade	Possui	Windows, Linux e versão genérica
<b>Apache ServiceMix</b>	Atualizada	Última atualização em Fevereiro/2012	Em atividade	Possui	Windows, Linux, Mac
<b>Apache Synapse Enterprise Service Bus</b>	Atualizada	Última atualização em Janeiro/2012	Não foram encontrados links para o fórum no site	Possui	Windows, Linux
<b>Mule ESB</b>	Atualizada	Última atualização em Janeiro/2012	Em atividade	Possui	Windows, Linux, Mac
<b>FuseSource</b>	Atualizada	Última atualização em Março/2012	Em atividade	Possui	Windows, Linux, Mac
<b>Petals ESB</b>	Atualizada	Última atualização em Fevereiro/2012	Em atividade	Possui	Windows, Linux

<b>WSO2 Enterprise Service Bus</b>	Atualizada	Última atualização em Novembro/2011	Em atividade	Possui	Windows, Linux
<b>UltraESB</b>	Atualizada	Última atualização em Março/2012	Em atividade	Possui	Windows, Linux

Vale ressaltar que além dos ESBs comerciais analisados, existem outros disponíveis no mercado, como, por exemplo, IBM WebSphere Enterprise Service Bus<sup>10</sup>, Progress Sonic ESB<sup>11</sup>, TIBICO<sup>12</sup>, Software AG WebMethods Integration Server<sup>13</sup>, entre outros. Estes ESBs não foram avaliados por limitações de tempo.

Em abril de 2011, o *Forrester Research* apresentou uma análise das ferramentas ESB existentes no mercado [Vollmer *et al.*, 2011]. O FuseSource se destacou como sendo a melhor ferramenta dentre as de código aberto, perdendo apenas para ferramentas comerciais, entre eles o Software AG Web Methods, líder da análise. A Figura 1 apresenta a classificação das ferramentas.

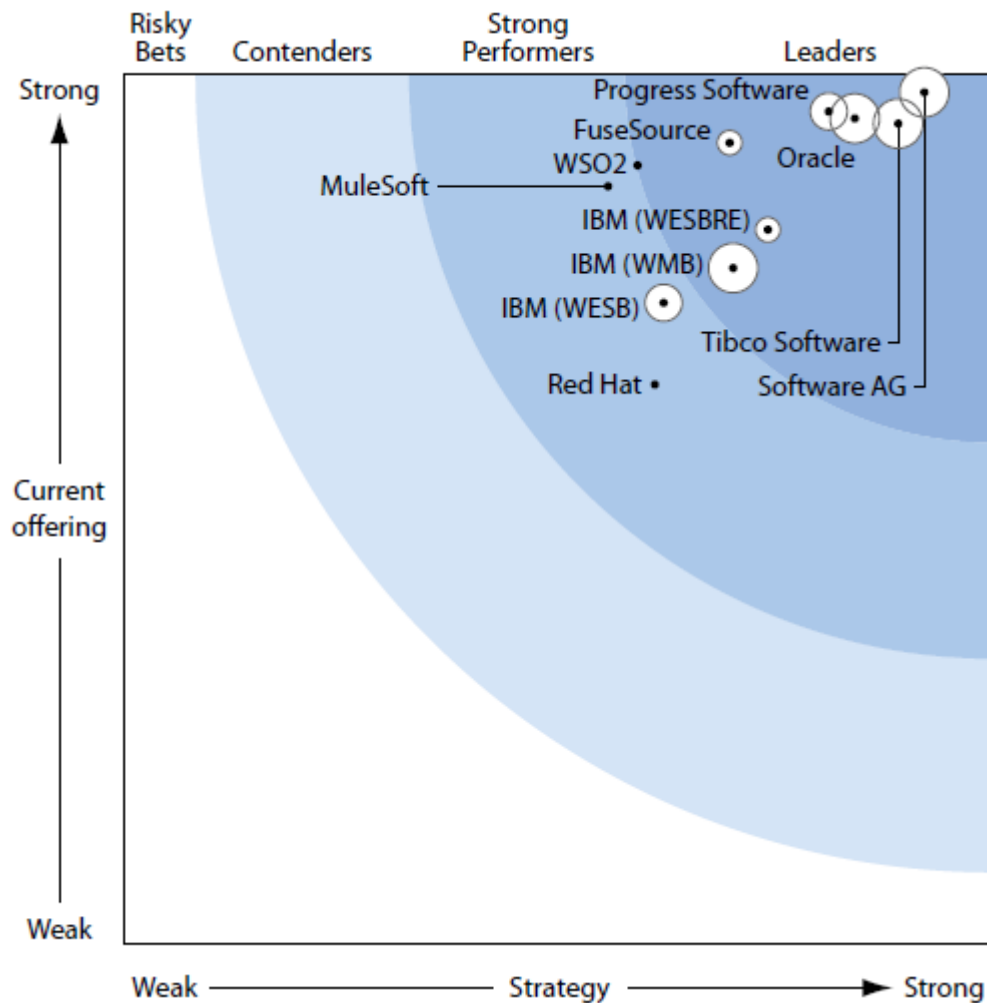
---

<sup>10</sup> <http://www-01.ibm.com/software/integration/wsesb/#>

<sup>11</sup> <http://www.progress.com/pt-br/sonic/sonic-esb.html>

<sup>12</sup> <http://www.tibco.com.br/solutions/soa/tibco-solutions/default.jsp>

<sup>13</sup> [http://www.softwareag.com/corporate/products/wm/soagovernance/webmethods\\_esb/overview/default.asp](http://www.softwareag.com/corporate/products/wm/soagovernance/webmethods_esb/overview/default.asp)



**Figura 1 - Avaliação dos ESB pela Forrester Research, Inc. (Vollmer *et al.*, 2011)**

Com base nos dados e características apresentados nesta seção, foram selecionadas 3 ferramentas: FuseSource (Fuse ESB), WSO2 e Oracle.

Primeiro foi instalado o Fuse ESB, porém tivemos dificuldades em utilizá-lo, visto que sua execução é em linhas de comando, o que não torna a ferramenta intuitiva de utilizar, sendo necessário um longo estudo da documentação para sua utilização.

Considerando as dificuldades na utilização da ferramenta, a quantidade de documentação a ser estudada e o curto período de tempo disponível, partimos para a análise da segunda ferramenta, o WSO2. Nesta ferramenta foi possível avançar um pouco mais, realizando inclusive o cadastramento dos serviços. Porém, obtivemos problemas ao exportar o serviço que seria consumido pela aplicação cliente. Com

foco neste problema, foi criado um tópico no fórum da ferramenta, citando detalhes do erro, mas nenhuma resposta obtida serviu como solução.

Devido ao curto período de tempo, adicionado ao fato de não obter solução para os problemas encontrados decidimos por instalar a terceira ferramenta selecionada, o Oracle Service Bus, onde realizamos o trabalho.

O Oracle Service Bus é considerado um componente chave do Oracle SOA Suite, visto que amplia a performance e a escalabilidade deste. Por este motivo, a Oracle recomenda aos usuários do Oracle SOA Suite a instalação do Oracle Service Bus.

Para instalar o Oracle Service Bus é necessário instalar os seguintes softwares primeiro (pré-requisitos): Oracle WebLogic Server 11gR1 (10.3.6) + Coherence + OEPE - Package Installer. Depois de instalados, estes pré-requisitos, deve-se instalar Oracle Service Bus versão 11gR1 (11.1.1.6.0). O download de ambas as instalações está disponível em <http://www.oracle.com/technetwork/middleware/service-bus/downloads/index.html>. Para instalá-los basta seguir as orientações exibidas ao longo da mesma.

Maiores informações sobre a ferramenta podem ser obtidas a partir do link <http://www.oracle.com/technetwork/middleware/service-bus/overview/index.html> e no Apêndice 6 é apresentado um passo a passo para a criação de um domínio.

## CAPÍTULO 4: APLICAÇÃO DO MÉTODO

---

O objetivo deste capítulo é apresentar a aplicação dos métodos para identificação, análise, projeto e implementação de serviços propostos por [Azevedo *et al.*, 2009a, 2009b, 2011] e Dirr *et al.* [2012], apresentados no Capítulo 2, no modelo de processos referente à imobiliária OnLine descrito na seção 4.1.

### 4.1 Cenário utilizado na avaliação do trabalho

Este trabalho será focado e aplicado em um cenário fictício, o qual foi utilizado nas aulas de SOA da disciplina TABD2 do bacharelado em Sistemas de Informação da UniRio, no segundo semestre de 2011.

O cenário corresponde à compra e venda de imóveis pela internet considerando a parceria de diversas imobiliárias. Quando o cliente acessa o portal, ele tem acesso não só aos imóveis da imobiliária de origem, como também aos imóveis de outras imobiliárias. Caso o imóvel escolhido pelo cliente seja de outra imobiliária, a comissão é dividida em duas partes, sendo cada parte de uma imobiliária. Portanto, parte da transação é realizada com a imobiliária de origem e parte com a outra imobiliária. O cálculo da comissão é feito de forma diferenciada entre cada imobiliária parceira. A Figura 2 ilustra a integração entre as diferentes imobiliárias.

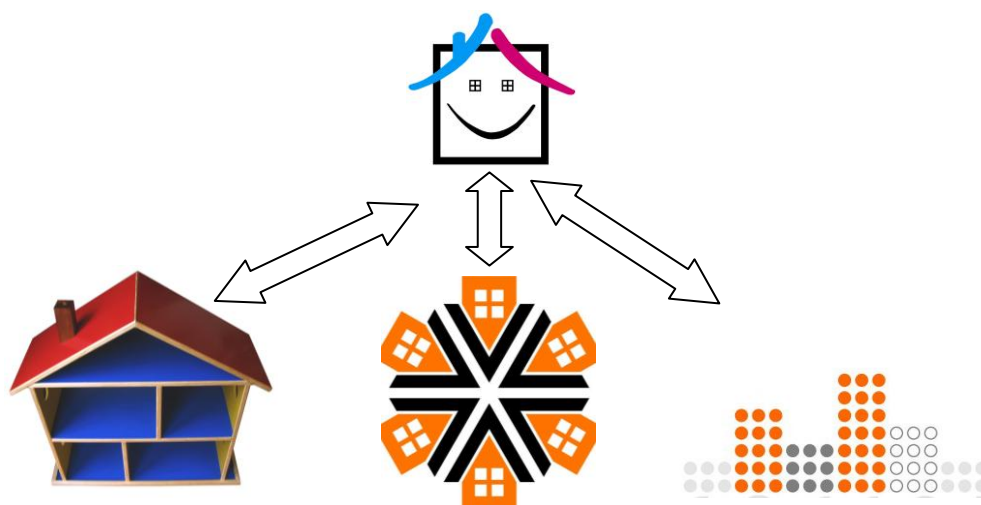
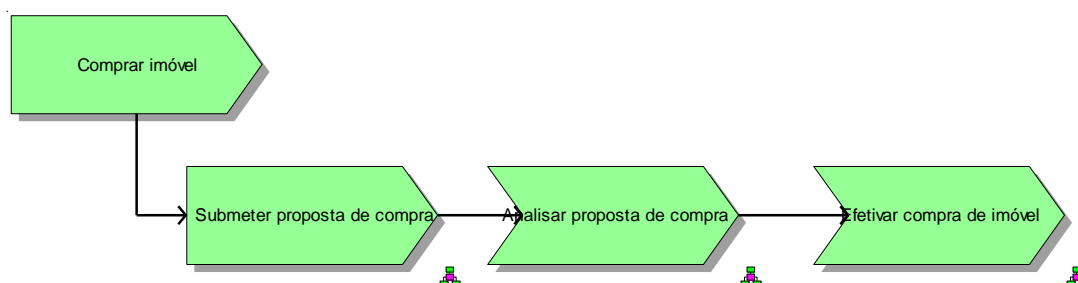


Figura 2 - Integração entre as diferentes imobiliárias

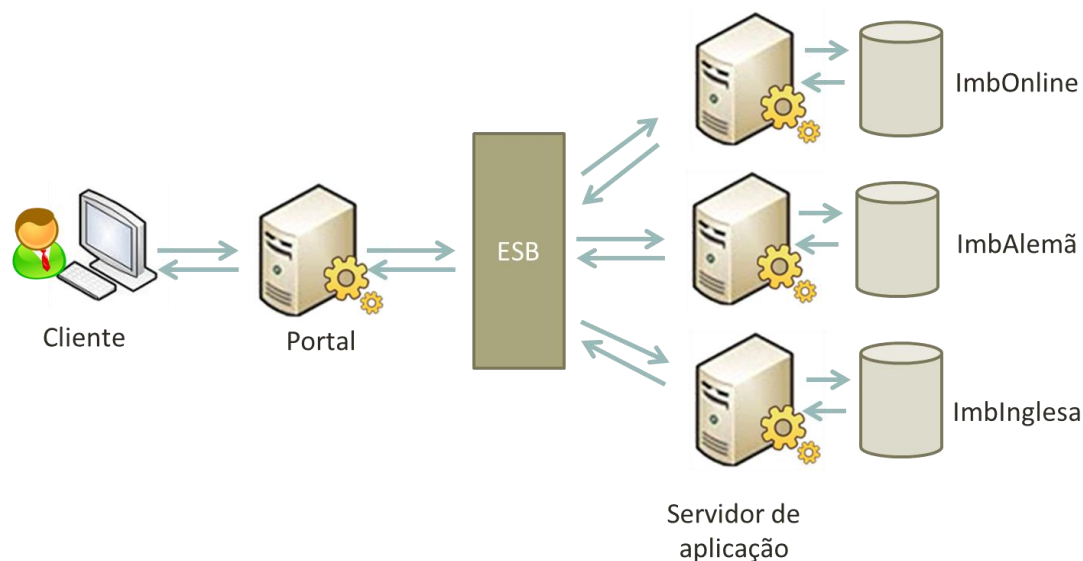
Logo, a partir do cenário descrito acima foi elaborado o diagrama de macroprocesso “Comprar imóvel” apresentado na Figura 3. Este trabalho foca no processo “Submeter proposta de compras” cujos detalhes são apresentados no Apêndice 1.



**Figura 3 - Diagrama do macro processo "Comprar imóvel"**

Vale ressaltar que este processo é um processo novo, e, portanto, tanto a Imobiliária Online como suas parceiras não possuem nenhuma implementação. Com isso, observou-se que a abordagem SOA seria a mais adequada para a integração entre as imobiliárias, uma vez que as imobiliárias se encontram em locais físicos distintos e poderiam implementar seus sistemas em linguagens diferentes de acordo com suas necessidades. Além disso, novas oportunidades de integração com outras imobiliárias poderiam surgir e a abordagem SOA facilitaria essa integração de forma rápida e pouco custosa.

A partir destas considerações o cenário técnico para integração das imobiliárias foi elaborado e é apresentado na Figura 4. O ESB é a principal tecnologia deste cenário, e tem como objetivo integrar os serviços das diversas imobiliárias e disponibilizá-los para serem utilizados em um portal, o qual é utilizado pelo cliente final do sistema. Cada imobiliária possui seus imóveis armazenados em um banco dados e estes são consultados por um servidor de aplicação de acordo com as requisições feitas pelo ESB. O cliente utiliza um browser para acessar o portal que é disponibilizado em um servidor de aplicação. O portal se comunica com os serviços disponibilizados no ESB realizando acesso aos imóveis das imobiliárias de acordo com as requisições do cliente.



**Figura 4 - Arquitetura utilizada**

## 4.2 Identificação

Esta seção apresenta as heurísticas de identificação de serviços candidatos e o resultado de aplicação das mesmas no processo de negócio “Submeter proposta de compra”.

### 4.2.1 Identificação dos serviços

Esta etapa consiste na identificação dos serviços candidatos a partir do modelo de processos de negócio. Os serviços identificados serão apresentados de acordo com o *template* da Tabela 2.

**Tabela 2 - Template para identificação dos serviços**

Identificador do Serviço Candidato <corresponde ao nome atribuído ao serviço>	
Tipo	<indicação do tipo do serviço candidato: lógica, dados, utilitário>
Entrada	<informações de entrada>
Saída	<informações de saída>
Origem	<elemento do qual o serviço foi identificado ou, no caso de serviço identificado a partir de fluxo, nome da heurística utilizada na identificação do serviço>
Atividade	<atividades que foram consideradas para



	identificação deste serviço candidato>
Descrição	<breve descrição do serviço candidato>
Observação	<observação relevante sobre o serviço identificado>

O resultado dos serviços identificados, que satisfazem as heurísticas, são apresentados abaixo de forma agrupada:

#### 4.2.1.1 Heurística de identificação de serviços a partir de regra de negócio

**"Heurística: Um serviço candidato deve ser identificado a partir de uma regra de negócio."**

Nesta heurística devem-se observar as regras de negócio presentes no modelo de processos de negócio, obtendo-se um serviço candidato referente a cada uma destas. O serviço candidato identificado a partir de regras de negócio está expresso na Tabela 3.

**Tabela 3 - Heurística de regra de negócio**

Listar imobiliárias por bairro	
Tipo	Lógica
Entrada	Bairro
Saída	Imobiliárias parceiras que atendem ao bairro, correspondente ao cluster "Imobiliárias parceiras"
Origem	Regra de negócio "Abrangência de imobiliária parceira"
Atividade	Analisar critérios de pesquisa
Descrição	A abrangência de uma imobiliária parceira é definida pelos bairros que ela possui imóvel.

#### **4.2.1.2 Heurística de identificação de serviços a partir de requisito de negócio**

**"Heurística: Um serviço candidato deve ser identificado a partir de um requisito de negócio."**

Nesta heurística devem-se observar os requisitos de negócio presentes no modelo de processos de negócio, obtendo-se um serviço candidato para cada requisito identificado. Os serviços candidatos identificados por esta heurística são exibidos na Tabela 4.

**Tabela 4 - Heurística de requisitos de negócio**

Serviço candidato	Registrar critérios de pesquisa	Verificar abrangência de imobiliárias parceiras	Consultar imóveis de imobiliárias parceiras	Consultar imóveis da imobiliária Online	Listar imóveis	Reservar imóvel	Registrar proposta de compra
Tipo	Dados	Dados	Dados	Dados	Dados	Dados	Dados
Entrada	Informações presentes no cluster “Critérios de pesquisa”	Bairro	Informações correspondentes ao cluster “Critérios de pesquisa”	Informações correspondentes ao cluster “Critérios de pesquisa”	Imóveis retornados pelas imobiliárias parceiras e pela Imobiliária Online, correspondendo às informações dos clusters “Imóveis disponíveis imobiliárias parceiras” e “Imóveis disponíveis”.	Imóvel escolhido pelo cliente, correspondente às informações do cluster “Imóvel escolhido”	Informações correspondentes ao cluster “Proposta de compra”
Saída		Imobiliárias parceiras que atendem ao bairro, correspondente às informações do cluster “Imobiliárias parceiras”	Lista de imóveis que atendem ao critério de pesquisa, contendo as informações correspondentes ao cluster “Imóveis disponíveis imobiliária parceira”	Lista de imóveis que atendem aos critérios de pesquisa, contendo as informações do cluster “Imóveis disponíveis”	Lista de imóveis a ser apresentada para o cliente.	Imóvel reservado ou não	Proposta de compra registrada ou não.
Origem	Requisito do negócio “Registrar critérios de pesquisa por imóveis”	Requisito do negócio “Verificar abrangência de imobiliárias parceiras”	Requisito do negócio “Consultar imóveis”	Requisito do negócio “Consultar imóveis disponíveis”	Requisito do negócio “Listar imóveis”	Requisito do negócio “Reservar imóvel”	Requisito do negócio “Registrar proposta de compra”
Atividade	Pesquisar imóvel	Analisar critérios de pesquisa	Obter imóveis disponíveis em parceiras	Obter imóveis disponíveis	Escolher imóvel	Escolher imóvel	Submeter proposta de compra
Descrição	O sistema Imobiliária Online deve registrar os critérios do cliente utilizados para pesquisar	O sistema Imobiliária Online deve identificar as imobiliárias parceiras que possam atender aos critérios de pesquisa do cliente de acordo com os bairros no qual a	O sistema da Imobiliária parceira deve consultar os imóveis disponíveis de acordo com os critérios de pesquisa, retornando:	O sistema Imobiliária Online deve consultar os imóveis disponíveis de acordo com os critérios de pesquisa, retornando:	O sistema Imobiliária Online deve listar os imóveis disponíveis, exibindo: - Endereço - Preço - Numero de quartos	O sistema Imobiliária Online deve reservar o imóvel escolhido pelo cliente o que pode ocorrer na própria Imobiliária ou em imobiliária	O sistema Imobiliária Online deve registrar a proposta de compra contendo: - Nome do cliente - CPF do cliente

	por imóveis disponíveis: - CPF do cliente - Endereço - Faixa de preço - Número de quartos - Se possui vaga de garagem - Se é sol da manhã e/ou da tarde	imobiliária possui imóveis.	- Endereço - Preço - Número de quartos - Se possui vaga de garagem - Se é sol da manhã e/ou da tarde - Fotos do imóvel	- Endereço - Preço - Numero de quartos - Se possui vaga de garagem - Se é sol da manhã e/ou da tarde - Fotos do imóvel	- Se possui vaga de garagem - Se é sol da manhã e/ou da tarde - Fotos do imóvel	parceira.	- Telefone - Endereço - Imóvel escolhido - Imobiliária - Data da proposta
--	---	-----------------------------	---	---	---	-----------	---

#### 4.2.1.3 Heurística de identificação de serviços a partir de *cluster*

"Heurística: Um serviço candidato de dado deve ser identificado para cada cluster de entrada de uma atividade (o qual também deve ser marcado como serviço de leitura), assim como serviço candidato de dado deve ser identificado para cada cluster de saída de uma atividade (o qual deve ser marcado como de escrita), desde que os clusters estejam associados a portadores de informação."

Para esta heurística, devem-se identificar serviços candidatos de dados para cada cluster (entrada e saída) presente no modelo de processos de negócio. Vale ressaltar que o cluster de saída de uma atividade só deve ser levado em conta caso esteja associado a algum portador de informação.

Os serviços candidatos referentes a um cluster de entrada devem ser marcados como serviços de leitura, enquanto os serviços candidatos referentes a um cluster de saída devem ser marcados como serviços de escrita. Estes serviços podem ser vistos na Tabela 5.

**Tabela 5 - Heurística de informações de entrada e saída**

Serviço candidato	Identificar imobiliárias parceiras por bairro	Listar imóveis de imobiliárias parceiras	Listar imóveis da imobiliária Online	Efetuar proposta de compra
Tipo	Dados (leitura)	Dados (leitura)	Dados (leitura)	Dados (escrita)
Entrada	Bairro	Informações correspondentes ao cluster “Critérios de pesquisa”	Informações correspondentes ao cluster “Critérios de pesquisa”	Informações correspondentes ao cluster “Proposta de compra”
Saída	Imobiliárias parceiras que atendem ao bairro, correspondente às informações do cluster “Imobiliárias parceiras”	Lista de imóveis de acordo com a entrada, correspondente ao cluster “Imóveis imobiliária parceira”	Lista de imóveis de acordo com a entrada, correspondente ao cluster “Imóveis disponíveis”	Proposta de compra registrada ou não.
Origem	Cluster “Imobiliárias parceiras”	Cluster “Imóveis imobiliária parceira”	Cluster “Imóveis”	Cluster “Proposta de compra”
Atividade	Analisar critérios de pesquisa	Obter imóveis disponíveis em parceiras	Obter imóveis disponíveis	Submeter proposta de compra
Descrição	O sistema Imobiliária Online identifica as imobiliárias parceiras que podem ter imóveis disponíveis para serem listados.	É feita uma busca por imóveis na base de dados das imobiliárias parceiras de acordo como os critérios de pesquisa informados.	A Imobiliária Online deve consultar os imóveis que possui de acordo com os critérios de pesquisa.	A partir do imóvel escolhido, o Cliente cadastra sua proposta de compra.

#### **4.2.1.4 Heurística de identificação de serviços a partir de atividades sequenciais**

**"Heurística: Um serviço candidato deve ser identificado para um conjunto de atividades sequenciais."**

Nesta heurística deve-se observar as atividades do modelo de processo de negócio de forma a identificar um serviço candidato referente a cada caso onde duas ou mais atividades sejam executadas de forma sequencial. Os serviços candidatos identificados por esta heurística são mostrados na Tabela 6.



**Tabela 6 - Heurística de atividades sequenciais**

Serviço candidato	Pesquisar imobiliárias por bairro	Efetivar proposta de compra
Tipo	Dados	Dados
Entrada	Bairro	Imóveis disponíveis da imobiliária online e/ou imobiliária parceira, correspondente aos clusters “Imóveis disponíveis imobiliárias parceiras” e “Imóveis”
Saída	Imobiliárias que atendem ao bairro, contendo informações correspondentes ao cluster “Imobiliárias parceiras com disponibilidade”	Proposta de compra cadastrada ou não.
Origem	Atividades sequenciais	Atividades sequenciais
Atividade	- Pesquisar imóveis - Analisar critérios de pesquisa	- Escolher imóvel - Submeter proposta de compra
Descrição	Realizar a pesquisa a partir da análise critérios de pesquisa.	A partir da escolha dos imóveis cadastrar a proposta de compra e, em seguida, submetê-la para o processo “Analisar proposta de compra”.

#### 4.2.1.5 Heurística de identificação de serviços a partir de operadores lógicos *OR*

"Heurística: Um serviço candidato deve ser identificado a partir de uma estrutura iniciada em um ponto no *workflow* onde, baseado em uma decisão, uma ou mais ramificações do fluxo é escolhida e finalizada em um ponto no *workflow* onde as várias ramificações do fluxo se juntem. Se mais de um dos fluxos de origem for executado, então é necessário sincronizá-los. Ramificações podem também terminar em um evento final."

Nesta heurística deve-se observar o modelo de processos de negócio em busca de operadores lógicos, identificando-se um serviço candidato que envolva o contexto deste operador (atividade anterior ao operador e suas ramificações). O serviço candidato identificado por esta heurística pode ser visto na Tabela 7.

**Tabela 7 - Heurística de OR**

<b>Buscar imóveis</b>	
Tipo	Dados
Entrada	Informações correspondentes ao cluster "Critérios de pesquisa"
Saída	Imóveis disponíveis da imobiliária online e/ou imobiliária parceira, correspondente aos clusters "Imóveis disponíveis imobiliárias parceiras" e "Imóveis disponíveis"
Origem	Heurística de OR
Atividade	<ul style="list-style-type: none"><li>- Analisar critérios de pesquisa</li><li>- Obter imóveis disponíveis</li><li>- Obter imóveis disponíveis em parceiras</li></ul>
Descrição	Tem como objetivo a partir de um critério de pesquisa obter os imóveis próprios e imóveis de parceiras.

#### 4.2.1.6 Heurística de identificação de serviços a partir de interfaces de processo

"Heurística: Um serviço candidato deverá ser identificado a partir de uma interface que inicia um processo desde que em pelo menos um dos caminhos a partir da interface a primeira atividade seja automatizada. Este serviço é responsável por receber a mensagem do outro processo e iniciar a atividade automatizada. Da mesma forma, um serviço candidato deverá ser identificado a partir de uma interface que finaliza um processo desde que em pelo menos um dos caminhos que chegam na interface a atividade exatamente anterior seja automatizada. Este serviço é responsável por enviar a mensagem para o outro processo."

Nesta heurística devem-se observar as interfaces de processo presentes no modelo de processo de negócio, referenciando a ela um serviço candidato quando:

- Interface inicial de um processo – A primeira atividade de pelo menos um dos caminhos a partir da interface deve ser automatizada.
- Interface final de um processo – Pelo menos um dos caminhos que chegam até a interface deve ter sua última atividade automatizada, de forma que esta anteceda a interface.

O serviço candidato identificado por esta heurística é mostrado na Tabela 8

**Tabela 8 - Heurística interface de processos**

Submeter proposta de compra	
Tipo	Dados
Entrada	Informações correspondentes ao cluster "Proposta de compra"
Saída	Proposta de compra submetida
Origem	Heurística interface de processos
Atividade	Submeter proposta de compra
Descrição	A proposta de compra do cliente é submetida para o processo "Analisar proposta de compra".

#### 4.2.1.7 Heurística de identificação de serviços a partir de atividades de múltiplas instâncias

"Heurística: Os seguintes serviços candidatos devem ser identificados a partir de um ponto do *workflow* onde múltiplas mensagens são enviadas a partir de uma atividade para outras atividades conhecidas:

- Um serviço candidato de dado deve ser identificado para o remetente da mensagem. Este serviço será responsável por transmitir as informações para os serviços destinatários.
- Um serviço candidato deve ser identificado para cada destinatário.
- Um serviço candidato de dado deve ser identificado para consolidar as execuções das diferentes instâncias da atividade de múltiplas instâncias e transferi-la para o destinatário."

Nesta heurística deve-se localizar no workflow uma atividade de onde sejam enviadas múltiplas mensagens para outras atividades conhecidas. Desta forma, um serviço candidato deve ser identificado para o remetente da mensagem, assim como para cada destinatário. Além destes, um serviço candidato de dados deve ser identificado com o objetivo de consolidar as execuções da atividade de múltiplas instâncias. Os serviços candidatos identificados por esta heurística podem ser vistos na Tabela 9.

**Tabela 9 - Heurística atividade de múltipla instancia**

Serviço candidato	Invocar serviços de imobiliárias parceiras	Pesquisar imóveis	Consolidar resultado da pesquisa de imóveis
Tipo	Dados	Dados	Dados
Entrada	Imobiliárias parceiras com disponibilidade	Informações correspondentes ao cluster “Critérios de pesquisa”	Informações correspondentes ao cluster “Imóveis disponíveis imobiliárias parceiras”
Saída	Informações correspondentes ao cluster “Critérios de pesquisa”	Imóveis disponíveis de imobiliária parceira, correspondentes ao cluster “Imóveis disponíveis imobiliária parceira”	Conjunto de imóveis disponíveis em uma estrutura única
Origem	Heurística atividade de múltiplas instâncias	Heurística atividade de múltiplas instâncias	Heurística atividade de múltiplas instâncias
Atividade	Obter imóveis disponíveis em parceiras (serviço candidato de dados)	Obter imóveis disponíveis em parceiras (serviço candidato)	Obter imóveis disponíveis em parceiras (serviço candidato)
Descrição	Tem como objetivo invocar os serviços de cada imobiliária parceira a fim de obter os imóveis disponíveis de cada uma de acordo com os critérios de pesquisa.	Corresponde ao serviço disponibilizado por cada imobiliária parceira para consultar os imóveis que elas têm disponíveis.	Tem como objetivo consolidar os resultados obtidos com a invocação dos serviços de cada imobiliária parceira para obter os imóveis disponíveis das mesmas.

## 4.2.2 Consolidação de serviços candidatos

Esta etapa tem como objetivo consolidar os serviços candidatos identificados pela etapa de Identificação, de forma a organizar e detalhar cada um deles. A Consolidação, assim como a Identificação, segue algumas heurísticas para auxiliar no processo.

### 4.2.2.1 Heurística de eliminação de serviços candidatos

**"Heurística: Devem ser eliminados todos os serviços candidatos que apareçam duplicados ou que refiram a regras que explicitam valores padrão."**

Nesta heurística, devem-se eliminar serviços duplicados ou que façam referência a regras que explicitem valores comuns, ou seja, caso dois serviços candidatos possuam o mesmo objetivo e façam referência à elementos do modelo de processo de negócio que sejam semelhantes, um deles deve ser eliminado para evitar redundância.

A Tabela 10 mostra na coluna esquerda os serviços candidatos identificados agrupados entre si de acordo com suas semelhanças. Na coluna direita estão os serviços candidatos restantes do processo de eliminação de serviços duplicados.

**Tabela 10 - Eliminação de serviços duplicados**

Serviços identificados	Serviços filtrados
1- Listar imobiliárias por bairro 2- Verificar abrangência de imobiliárias parceiras 3- Identificar imobiliárias parceiras por bairro 4- Pesquisar imobiliárias por bairro	1- Verificar abrangência de imobiliárias parceiras
1- Listar imóveis da imobiliária Online 2- Consultar imóveis da imobiliária Online	1- Listar imóveis da imobiliária Online
1- Listar imóveis de imobiliárias parceiras 2- Consultar imóveis de imobiliárias parceiras	1- Listar imóveis de imobiliárias parceiras
1- Buscar imóveis	1 – Buscar imóveis
1- Registrar critérios de pesquisa	1- Registrar critérios de pesquisa
1- Listar imóveis	Não virará serviço físico, pois corresponde a um requisito a ser

	implementado na interface do sistema que disponibilizará os imóveis para o cliente.
1- Reservar imóvel	1- Reservar imóvel
1- Registrar proposta de compra 2- Efetuar proposta de compra	1- Registrar proposta de compra
1- Efetivar proposta de compra	Este serviço candidato representa uma composição muito simples, correspondente a duas atividades, e envolve interação humana, não fazendo sentido implementá-lo como um serviço ou composição de serviços.
1- Submeter proposta de compra	1- Submeter proposta de compra
1- Invocar serviços de imobiliárias parceiras	1- Invocar serviços de imobiliárias parceiras
1- Pesquisar imóveis	Este serviço basicamente executa a mesma funcionalidade de pesquisa por imóveis de imobiliárias parceiras correspondentes aos serviços “Listar imóveis de imobiliárias parceiras” e “Consultar imóveis de imobiliárias parceiras”. Logo, este serviço foi removido dado que representa uma funcionalidade tratada pelo serviço “Listar imóveis de imobiliárias parceiras”.
1- Consolidar resultado da pesquisa de imóveis	1- Consolidar resultado da pesquisa de imóveis

#### **4.2.2.2 Listagem de serviços**

Após a eliminação de alguns serviços candidatos, tem-se a listagem com os serviços remanescentes. Na Tabela 11 podem ser vistos os serviços que serão submetidos às próximas heurísticas da etapa de Consolidação. A Tabela 12 exhibe as informações completas destes serviços.



**Tabela 11 - Listagem dos serviços candidatos**

<b>Serviços candidatos</b>
1- Registrar critérios de pesquisa
2- Verificar abrangência de imobiliárias parceiras
3- Listar imóveis da imobiliária Online
4- Listar imóveis de imobiliárias parceiras
5- Buscar imóveis
6- Invocar serviços de imobiliárias parceiras
7- Consolidar resultado da pesquisa de imóveis
8- Reservar imóvel
9- Registrar proposta de compra
10- Submeter proposta de compra

**Tabela 12 - Listagem completa dos serviços candidatos**

Serviço candidato	Registrar critérios de pesquisa	Verificar abrangência de imobiliárias parceiras	Listar imóveis da imobiliária Online	Listar imóveis de imobiliárias parceiras	Buscar imóveis	Invocar serviços de imobiliárias parceiras	Consolidar resultado da pesquisa de imóveis	Reservar imóvel	Registrar proposta de compra	Submeter proposta de compra
Tipo	Dados	Dados	Dados	Dados	Dados	Dados	Dados	Dados	Dados	Dados
Entrada	Critérios de pesquisa para a consulta de imóveis com os seguintes	Bairro	Critérios de pesquisa para a consulta de imóveis com os seguintes	Critérios de pesquisa para a consulta de imóveis com os seguintes	Critérios de pesquisa para a consulta de imóveis com os seguintes	Imobiliárias parceiras com disponibilidade	Imóveis disponíveis nas imobiliárias parceiras que atendem aos	Imóvel selecionado pelo cliente, contendo identificador do	Proposta de compra do cliente contendo: - informações	Proposta de compra do cliente contendo: - Informações

	campos: - Endereço; - Faixa de preço; - Número de quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.		campos: - Endereço; - Faixa de preço; - Número de quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.	campos: - Endereço; - Faixa de preço; - Número de quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.	campos: - Endereço; - Faixa de preço; - Número de quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.		critérios fornecidos, contendo: - Informações das imobiliárias parceiras: - Nome; - CNPJ; - Endereço; - Telefones para contato; - Abrangência; - Informações dos imóveis, contendo para cada imóvel: - Identificador do imóvel; - Endereço; - Preço; - Número de quartos; - Se possui vaga na garagem; - Se é sol da manhã e/ou sol da tarde; - Área; - As fotos do imóvel; - Outros detalhes do imóvel.	imóvel e CNPJ da imobiliária ao qual o imóvel pertence.	do cliente: - Nome do cliente; - CPF do cliente; - Telefone; - Endereço; - E-mail. - Informações do imóvel escolhido, contendo: - CNPJ da imobiliária ao qual o imóvel pertence; - Identificador do imóvel; - Endereço; - Preço; - Número de quartos; - Se possui vaga na garagem; - Se é sol da manhã e/ou sol da tarde; - Valor proposto pelo cliente para a compra do imóvel; - Data da proposta; - Área.	do cliente: - Nome do cliente; - CPF do cliente; - Telefone; - Endereço; - E-mail. - Informações do imóvel escolhido, contendo: - CNPJ da imobiliária ao qual o imóvel pertence; - Identificador do imóvel; - Endereço; - Preço; - Número de quartos; - Se possui vaga na garagem; - Se é sol da manhã e/ou sol da tarde; - Valor proposto pelo cliente para a compra do imóvel. - Data da proposta - Área
Saída		Lista de imobiliárias parceiras e informações de seus imóveis disponíveis a partir do bairro pesquisado,	Imóveis disponíveis da imobiliária Online, contendo: - Endereço; - Preço; - Número de	Imóveis disponíveis das imobiliárias parceiras, contendo: - Informações das imobiliárias parceiras:	Imóveis disponíveis nas imobiliárias parceiras que atendem aos critérios fornecidos, contendo:	Critérios de pesquisa para a consulta de imóveis com os seguintes campos: - Endereço; - Faixa de preço; - Número de	Conjunto de imóveis disponíveis em uma estrutura única	Imóvel reservado ou não	Proposta de compra registrada ou não.	Proposta de compra submetida

		contendo os seguintes campos da imobiliária: - Nome da imobiliária; - CNPJ; - Endereço; - Telefones para contato.	quartos; - Se possui vaga na garagem; - Se é sol da manhã e/ou sol da tarde; - Área - As fotos do imóvel; - Outros detalhes do imóvel.	- Nome; - CNPJ; - Endereço; - Telefones para contato; - Informações dos imóveis, contendo para cada imóvel: - Identificador do imóvel; - Endereço; - Preço; - Número de quartos; - Se possui vaga na garagem; - Área - Se é sol da manhã e/ou sol da tarde; - As fotos do imóvel; - Outros detalhes do imóvel.	- Informações das imobiliárias parceiras: - Nome; - CNPJ; - Endereço; - Telefones para contato; - Abrangência; - Informações dos imóveis, contendo para cada imóvel: - Identificador do imóvel; - Endereço; - Preço; - Número de quartos; - Se possui vaga na garagem; - Se é sol da manhã e/ou sol da tarde; - Área; - As fotos do imóvel; - Outros detalhes do imóvel. Imóveis disponíveis da imobiliária Online, contendo: - Endereço; - Preço; - Número de quartos; - Se possui vaga na garagem; - Se é sol da	quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.				
--	--	---	---	---	---	--	--	--	--	--

					manhã e/ou sol da tarde; - Área; - As fotos do imóvel; - Outros detalhes do imóvel.					
Descrição	O sistema Imobiliária Online deve registrar os critérios do cliente utilizados para pesquisar por imóveis disponíveis: - CPF do cliente - Endereço - Faixa de preço - Número de quartos - Se possui vaga de garagem - Se é sol da manhã e/ou da tarde	O sistema Imobiliária Online deve identificar as imobiliárias parceiras que possam atender aos critérios de pesquisa do cliente de acordo com os bairros no qual a imobiliária possui imóveis.	A Imobiliária Online deve consultar os imóveis que possui de acordo com os critérios de pesquisa.	É feita uma busca por imóveis na base de dados das imobiliárias parceiras de acordo como os critérios de pesquisa informados.	Tem como objetivo a partir de um critério de pesquisa obter os imóveis próprios e imóveis de parceiras.	Tem como objetivo invocar os serviços de cada imobiliária parceira a fim de obter os imóveis disponíveis de cada uma de acordo com os critérios de pesquisa.	Tem como objetivo consolidar os resultados obtidos com a invocação dos serviços de cada imobiliária parceira para obter os imóveis disponíveis das mesmas.	O sistema Imobiliária Online deve reservar o imóvel escolhido pelo cliente o que pode ocorrer na própria Imobiliária ou em imobiliária parceira.	O sistema Imobiliária Online deve registrar a proposta de compra contendo: - Nome do cliente - CPF do cliente - Telefone - Endereço - Imóvel escolhido - Imobiliária	A proposta de compra do cliente é submetida para o processo “Analisar proposta de compra”.

#### 4.2.2.3 Identificar grau de reuso para cada serviço candidato

“Heurística:

- O grau de reuso de um serviço candidato identificado a partir de uma atividade corresponde à soma do número de ocorrências de cada atividade em que o serviço está relacionado (de acordo com o elemento do qual o serviço foi identificado).
- O grau de reuso de um serviço candidato identificado a partir de um conjunto de atividades (por exemplo, serviço candidato identificado a partir de atividades sequenciais) é igual ao número de ocorrências do conjunto de atividades na mesma ordem que estas foram definidas no serviço.
- Além disso, quando um serviço candidato “A” compõe outro serviço candidato “B”, então se deve somar o grau de reuso do serviço composto “B” ao grau de reuso do serviço “A”.”

“O serviço identificado a partir de atividades de múltiplas instâncias deve conter uma marcação especial a fim de indicar que o seu reuso é maior do que o calculado pela heurística de grau de reuso de serviço candidato, pois ele também é reutilizado por cada instância da atividade.”

Nesta heurística deve-se analisar a procedência de cada serviço candidato, de forma a verificar seu grau de reuso. O resultado da aplicação desta heurística é mostrado na Tabela 13.

**Tabela 13 - Identificação do grau de reuso para cada serviço candidato**

Serviços candidatos	Grau de reuso	Ocorrências dos elementos de processo a partir do qual os serviços foram identificados
Registrar critérios de pesquisa	1	Atividade “Pesquisar imóvel”.
Verificar abrangência de imobiliárias parceiras	1	Atividade “Analisar critérios de pesquisa”.
Listar imóveis da imobiliária Online	1	Atividade “Obter imóveis disponíveis”.
Listar imóveis de imobiliárias parceiras	n	Atividade “Obter imóveis disponíveis em parceiras”.
Buscar imóveis	1	Composição de serviços com as atividades “Analisar critérios de pesquisa, Obter imóveis disponíveis, Obter

		imóveis disponíveis em parceiras”.
Invocar serviços de imobiliárias parceiras	n	Atividade de múltiplas instâncias “Obter imóveis disponíveis em parceiras”.
Consolidar resultado da pesquisa de imóveis	1	Atividade “Obter imóveis disponíveis em parceiras”
Reservar imóvel	1	Atividade “Escolher imóvel”.
Registrar proposta de compra	1	Atividade “Submeter proposta de compra”.
Submeter proposta de compra	1	Atividade “Submeter proposta de compra”.

#### 4.2.2.4 Heurística de associação de serviços candidatos com sistemas

**“Heurística: Um serviço candidato identificado a partir de um requisito de negócio deve ser associado aos sistemas que o implementam na tabela de consolidação.”**

Nesta heurística deve-se verificar os sistemas que implementam os requisitos a partir dos quais os serviços candidatos foram identificados. Ou seja, a partir da associação entre Sistema e requisito deriva-se a associação entre sistema e serviço. O resultado desta heurística é exibido na Tabela 14.

**Tabela 14 - Associação de serviços candidatos com sistemas que os implementam**

<b>Serviços candidatos</b>	<b>Sistemas</b>
Registrar critérios de pesquisa	Imobiliária online
Verificar abrangência de imobiliárias parceiras	Imobiliária online
Listar imóveis da imobiliária Online	Imobiliária online
Listar imóveis de imobiliárias parceiras	Imobiliária parceira
Buscar imóveis	Imobiliária online e Imobiliária parceira
Invocar serviços de imobiliárias parceiras	Imobiliária online
Consolidar resultado da pesquisa de imóveis	Imobiliária online
Reservar imóvel	Imobiliária online
Registrar proposta de compra	Imobiliária online
Submeter proposta de compra	Imobiliária online

#### 4.2.2.5 Heurística de associação de serviços candidatos com requisitos da demanda

**"Heurística: Um serviço candidato identificado a partir de um requisito deve ser associado ao respectivo requisito da demanda."**

Para esta heurística, deve-se associar os serviços candidatos aos seus respectivos requisitos de demanda. Considerando que todos os requisitos fazem parte da demanda para o desenvolvimento de sistemas, todos os serviços identificados a partir de requisitos estarão associados aos requisitos da demanda. O resultado da aplicação desta heurística pode ser visto na Tabela 15.

**Tabela 15 - Associação de serviços candidatos aos requisitos da demanda**

Serviços candidatos	Requisitos da demanda
1- Registrar critérios de pesquisa	1- Registrar critérios de pesquisa
2- Verificar abrangência de imobiliárias parceiras	2- Verificar abrangência de imobiliárias parceiras
3- Listar imóveis da imobiliária Online	3- Listar imóveis da imobiliária Online
4- Listar imóveis de imobiliárias parceiras	4- Listar imóveis de imobiliárias parceiras
5- Reservar imóvel	5- Reservar imóvel
6- Registrar proposta de compra	6- Registrar proposta de compra

#### 4.2.2.6 Heurística de associação de serviços candidatos com papéis

**"Heurística: Um serviço candidato deve ser associado aos papéis que executam a(s) atividade(s) que lhe(s) deram origem."**

Nesta heurística deve-se observar os serviços candidatos a fim de se identificar os papéis associados a eles, ou seja, aqueles que executam a(s) atividade(s) de origem do serviço. O resultado desta heurística é mostrado na Tabela 16.

**Tabela 16 - Associação de serviços candidatos com papéis**

Serviços candidatos	Papéis
Registrar critérios de pesquisa	Cliente
Verificar abrangência de imobiliárias parceiras	
Listar imóveis da imobiliária Online	
Listar imóveis de imobiliárias parceiras	
Buscar imóveis	

Invocar serviços de imobiliárias parceiras	
Consolidar resultado da pesquisa de imóveis	
Reservar imóvel	Cliente
Registrar proposta de compra	Cliente
Submeter proposta de compra	Cliente

#### 4.2.2.7 Heurística de associação de serviços candidatos com atividades

**"Heurística: Um serviço candidato deve ser associado à(s) atividade(s) que lhe(s) deram origem."**

Nesta heurística deve-se associar os serviços candidatos com suas atividades de origem. A Tabela 17 exibe o resultado desta associação.

**Tabela 17 - Associação de serviços às atividades das quais eles foram identificados**

<b>Serviços candidatos</b>	<b>Atividades</b>
Registrar critérios de pesquisa	Pesquisar imóvel
Verificar abrangência de imobiliárias parceiras	Analisar critérios de pesquisa Pesquisar imóveis
Listar imóveis da imobiliária Online	Obter imóveis disponíveis
Listar imóveis de imobiliárias parceiras	Obter imóveis disponíveis em parceiras
Buscar imóveis	Analisar critérios de pesquisa Obter imóveis disponíveis Obter imóveis disponíveis em parceiras
Invocar serviços de imobiliárias parceiras	Obter imóveis disponíveis em parceiras
Consolidar resultado da pesquisa de imóveis	Obter imóveis disponíveis em parceiras
Reservar imóvel	Escolher imóvel
Registrar proposta de compra	Submeter proposta de compra
Submeter proposta de compra	Submeter proposta de compra

#### 4.2.2.8 Heurística de associação de serviços candidatos a partir de associações entre elementos do modelo

**"Heurística: Um serviço candidato identificado deve ser associado aos serviços candidatos que o utilizam. Esta informação é obtida a partir da associação entre regra de negócio × regra de negócio, regra de negócio × requisito de negócio, regra de negócio × cluster e requisito de negócio × cluster, quando esta está explícita na modelagem."**

Para esta heurística, deve-se buscar relações entre os serviços candidatos a partir da associação de seus elementos de origem (regras de negócio, requisitos de negócio e clusters) no modelo de processos de negócio.



Após a eliminação de serviços duplicados, não foram identificadas associações entre serviços candidatos a partir de associações de elementos do modelo de processos de negócio. Dessa forma, a Tabela 18 apenas ressalta a origem de cada serviço.

**Tabela 18 - Associação de serviços candidatos a partir da origem do serviço**

<b>Serviços candidatos</b>	<b>Origem</b>
Registrar critérios de pesquisa	Requisito do negócio <i>Registrar critérios de pesquisa por imóveis</i> da atividade <i>Pesquisar imóvel</i>
Verificar abrangência de imobiliárias parceiras	- Regra de negócio <i>Abrangência de imobiliária parceira</i> , requisito do negócio <i>Verificar abrangência de imobiliárias parceiras</i> e cluster <i>Imobiliárias parceiras</i> da atividade <i>Analisar critérios de pesquisa</i> - Atividades sequenciais <i>Pesquisar imóveis</i> e <i>Analisar critérios de pesquisa</i>
Listar imóveis da imobiliária Online	Cluster <i>Imóveis</i> e requisito do negócio <i>Consultar imóveis disponíveis</i> da atividade <i>Obter imóveis disponíveis</i>
Listar imóveis de imobiliárias parceiras	Cluster <i>Imóveis imobiliária parceira</i> e requisito do negócio <i>Consultar imóveis</i> da atividade <i>Obter imóveis disponíveis em parceiras</i>
Buscar imóveis	Heurística de OR das atividades: - <i>Analisar critérios de pesquisa</i> - <i>Obter imóveis disponíveis</i> - <i>Obter imóveis disponíveis em parceiras</i>
Invocar serviços de imobiliárias parceiras	Heurística de atividades de múltiplas instâncias da atividade <i>Obter imóveis disponíveis em parceiras</i>
Consolidar resultado da pesquisa de imóveis	Heurística de atividades de múltiplas instâncias da atividade <i>Obter imóveis disponíveis em parceiras</i>
Reservar imóvel	Requisito do negócio <i>Reservar imóvel</i> da atividade <i>Escolher imóvel</i>
Registrar proposta de compra	Requisito do negócio <i>Registrar proposta de compra</i> e cluster <i>Proposta de compra</i> da atividade <i>Submeter proposta de compra</i>
Submeter proposta de compra	Heurística de interface entre processos da atividade <i>Submeter proposta de compra</i>

## 4.3 Análise

Esta seção apresenta as heurísticas para análise de serviços candidatos e o resultado de aplicação das mesmas no processo de negócio “Submeter proposta de compra”.

### 4.3.1 Priorização de serviços candidatos

Esta etapa tem como objetivo destacar a importância de serviços candidatos. Para tal, é necessário que sejam atribuídos pesos para cada uma das informações a seguir, realizando uma ordenação de acordo com o somatório dos valores calculados.

#### 4.3.1.1 Grau de reuso de cada serviço candidato

**“Heurística: Agrupar serviços candidatos para cada diferente grau de reuso encontrado, listando grau de reuso e quantidade de serviços candidatos correspondente ao grau de reuso. Em seguida, definir pesos para cada agrupamento.”**

Segundo esta heurística, deve-se avaliar o grau de reuso de cada serviço, de forma que, quanto maior for o grau de reuso do serviço candidato, maior deve ser o peso atribuído a ele. Este peso é atribuído pelo analista SOA a partir do conhecimento que ele tem do negócio e das características de TI (Tecnologia de Informação) da organização. A aplicação desta heurística é mostrada na Tabela 19. Para este caso foi definido que serviços com grau de reuso 1 tem peso 1 e serviços com grau de reuso  $n$  tem peso 2.

**Tabela 19 - Agrupamento por grau de reuso de serviços candidatos**

<b>Grau de reuso</b>	<b>Quantidade de serviços candidatos</b>	<b>Pesos</b>
1	9	1
$n$	2	2

#### 4.3.1.2 Associação de serviços candidatos com sistemas

**“Heurística: Agrupar serviços candidatos que estão associados a cada sistema que os implementa, listando sistema e quantidade de serviços candidatos. Em seguida, definir pesos para cada agrupamento.”**

Segundo esta heurística, deve-se verificar a que sistemas o serviço candidato está associado. Portanto, caso o serviço candidato esteja associado a um sistema, deve-se atribuir um peso a ele referente à importância deste sistema para a organização. Caso o serviço candidato esteja associado a mais de um sistema, pode-se atribuir um novo peso para este agrupamento ou simplesmente realizar a soma dos pesos atribuídos para os sistemas envolvidos. Por último, caso o serviço candidato não possua nenhum sistema associado, então seu peso deve ser menor em relação aos outros. Este peso é atribuído pelo analista SOA a partir do conhecimento que ele tem do negócio e das características de TI (Tecnologia de Informação) da organização. A aplicação desta heurística pode ser vista na Tabela 20.

**Tabela 20 - Agrupamento por sistemas que implementam serviços candidatos**

Sistema	Quantidade de serviços candidatos	Pesos
Imobiliária online	8	2
Imobiliária parceira	1	2
(Nenhum)	1	1

#### **4.3.1.3 Aumentar peso de serviços candidatos identificados a partir de atividades de múltiplas instâncias**

**"Heurística: Atribuir maior peso para serviços candidatos identificados a partir de atividades de múltiplas instâncias."**

Segundo esta heurística, deve-se atribuir um peso adicional para os serviços candidatos que foram identificados a partir de atividades de múltiplas instâncias, visto que estes tendem a ser mais utilizados. A aplicação desta heurística é apresentada na Tabela 21.

**Tabela 21 - Serviços candidatos identificados a partir de atividades de múltiplas instâncias**

Serviço candidato	Pesos
Invocar serviços de imobiliárias parceiras	1
Consolidar resultado da pesquisa de imóveis	1
Listar imóveis de imobiliárias parceiras	1

#### **4.3.1.4 Associação de serviços candidatos com requisitos da demanda**

**"Heurística: Definir pesos para serviços candidatos associados aos requisitos da demanda. Definir maior peso para serviços**

**candidatos associados aos requisitos de uma demanda com maior prioridade."**

Esta heurística busca atribuir um peso adicional aos serviços candidatos que fazem parte dos requisitos da demanda. A aplicação desta heurística pode ser vista na Tabela 22.

**Tabela 22 - Agrupamento de serviços candidatos associados aos requisitos da demanda**

Serviços candidatos	Pesos
10	1

#### **4.3.1.5 Associação de serviços candidatos com papéis**

**"Heurística: Agrupar serviços candidatos que estão associados a cada papel, listando papel e quantidade de serviços candidatos. Em seguida, definir pesos para cada agrupamento."**

Para a aplicação desta heurística, é verificado se o serviço candidato é executado por um sistema e/ou por humanos. Caso o serviço candidato seja executado apenas por um sistema, este deve receber um peso maior por ser totalmente automatizado. Um peso intermediário deve ser atribuído aos serviços candidatos que sejam executados por ambos, e um menor peso para o caso onde o serviço seja executado apenas por uma pessoa. Esta heurística é aplicada na Tabela 23.

**Tabela 23 - Agrupamento de papéis que utilizam (executam) o serviço**

Papéis	Unidades de Negócio	Quantidade de serviços candidatos	Pesos
Cliente		4	2
Imobiliária Online		4	3
Imobiliária Parceira		2	3

#### **4.3.1.6 Cálculo da priorização de serviços candidatos**

**"Heurística: Calcular priorização de serviços candidatos de acordo de acordo com o somatório dos pesos definidos por cada heurística de priorização."**

Tendo definido os pesos para as informações mostradas acima, a Tabela 24 mostra o somatório destes, obtendo-se, assim, a prioridade correspondente. Terão maior prioridade os serviços que tiverem um maior valor no somatório. Para este

cenário, observando cada peso atribuído nas colunas, as colunas *Reuso*, *Atividades de Múltiplas Instâncias* e *Papéis* foram as que mais contribuíram para a diferenciação das prioridades dos serviços. Dessa forma, estas características são as mais relevantes para a organização.

**Tabela 24 - Cálculo de priorização de serviços**

Serviços candidatos	Peso					Prior
	Reuso	Assoc. Sist.	Múlt. Inst.	Deman da	Papéis	
Registrar critérios de pesquisa	1	2	0	1	2	6
Verificar abrangência de imobiliárias parceiras	1	2	0	1	3	7
Listar imóveis da imobiliária Online	1	2	0	1	3	7
Listar imóveis de imobiliárias parceiras	2	2	1	1	3	9
Buscar imóveis	1	0	0	1	2	4
Invocar serviços de imobiliárias parceiras	2	2	1	1	3	9
Consolidar resultado da pesquisa de imóveis	1	2	1	1	3	8
Reservar imóvel	1	2	0	1	2	6
Registrar proposta de compra	1	2	0	1	3	7
Submeter proposta de compra	1	2	0	1	2	6

#### **4.3.1.7 Detalhamento da priorização de serviços identificados a partir de fluxo**

**“Heurística: Priorizar serviços de fluxo de acordo com as informações levantadas para caracterização dos mesmos.”**

Nesta heurística, deve-se atribuir pesos para as seguintes informações da etapa de consolidação: número de atividades automatizadas; número de atividades apoiadas por sistema que compõe o fluxo; número de modelos onde o fluxo aparece; número de entidades do modelo de dados associadas aos clusters das atividades que compõem o fluxo; e número de sub-fluxos existentes no fluxo que gerou o serviço. A

prioridade de cada serviço de fluxo é definida pelo somatório destes pesos, como mostrado na Tabela 25.

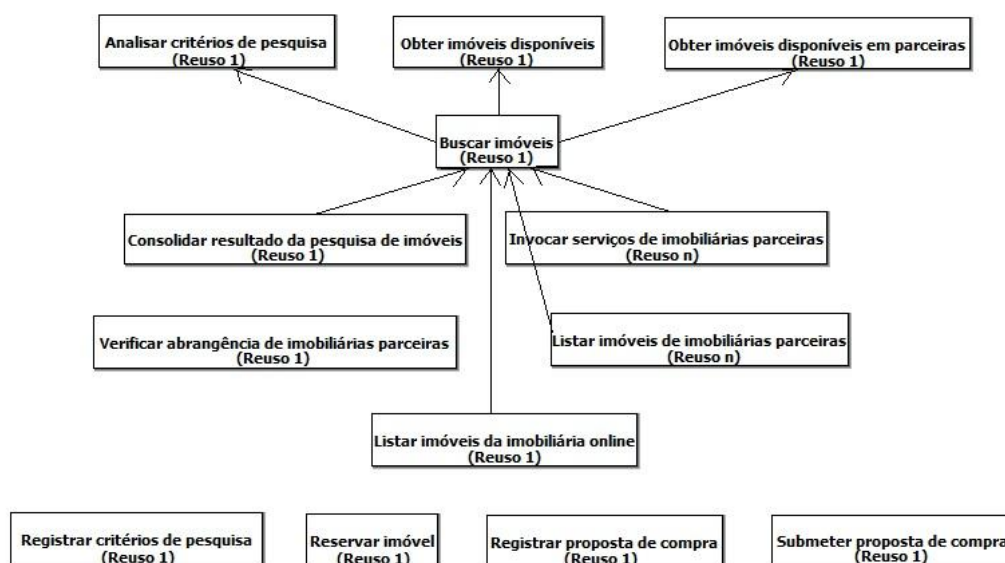
**Tabela 25 - Priorização de serviços identificados a partir de fluxo**

Serviços candidatos	Peso					Prior.
	#atv. Autom.	#atv. Apoiadas	#modelos	#entidades	#sub-fluxos	
Buscar imóveis	3	0	1	6	0	10

### 4.3.2 Definição da granularidade inicial de serviços candidatos

“Heurística: Elaborar mapa de granularidade de serviços candidatos e marcar serviços que têm maior ganho em disponibilizar como serviços físicos.”

Após a etapa de priorização dos serviços candidatos, iremos analisar a granularidade destes. Para esta etapa Azevedo *et al.* (2011) propõem o uso de uma proposta semelhante a de Marks e Bell *et al.* (2006), onde serviços mais finos devem ser colocados mais a baixo, enquanto serviços mais grossos devem ser colocados mais acima no mapa, sendo levada em consideração a dependência entre os serviços. Esta dependência deve ser identificada a partir do relacionamento entre os elementos que deram origem aos serviços candidatos. O mapa desenvolvido pode ser visto na Figura 5.



**Figura 5 - Mapa de granularidade**

O serviço “Buscar imóveis” corresponde a uma composição de serviços, incluindo os serviços: “Analisar critérios de pesquisa”, “Obter imóveis disponíveis”, “Obter imóveis disponíveis em parceiras”, “Consolidar resultado da pesquisa de imóveis”, “Verificar abrangência de imobiliárias parceiras”, “Invocar serviços de imobiliárias parceiras”, “Listar imóveis de imobiliárias parceiras” e “Listar imóveis da imobiliária online”.

### **4.3.3 Agrupamento de serviços candidatos**

**“Heurística: Associar serviços candidatos de dados com grupos de entidades indicando as operações CRUD (Create, Retrieve, Update e Delete) que o serviço realiza sobre entidades do grupo. Em seguida, agrupar serviços de acordo com estas operações.”**

**“Heurística: Definir canais de comunicação entre grupos de serviços candidatos de acordo com operações Read.”**

**“Heurística: Projetar serviços buscando reduzir o custo de comunicação entre eles.”**

O primeiro passo para esta etapa é identificar os grupos de entidades a partir do modelo canônico, como apresentado na Figura 6. Estes grupos (Cliente, Imobiliária e Imóvel), foram identificados a partir da análise do modelo canônico (ver Apêndice 2). Vale ressaltar que há um grupo que abrange as entidades Bairro, Cidade e Estado que pertence aos três grupos (Cliente, Imobiliária e Imóvel). Além disso, para que este passo possa ser executado é necessário que o modelo canônico já esteja elaborado. Em outras palavras a heurística de elaboração do modelo canônico deve ser executada antes da heurística de agrupamento de serviços candidatos. Neste caso, a heurística não precisa ser executada integralmente, pois apenas o conhecimento das classes já é suficiente para o entendimento dos grupos, ou seja, o modelo canônico para esta fase precisa ter, pelo menos, os nomes das entidades e os relacionamentos entre elas.

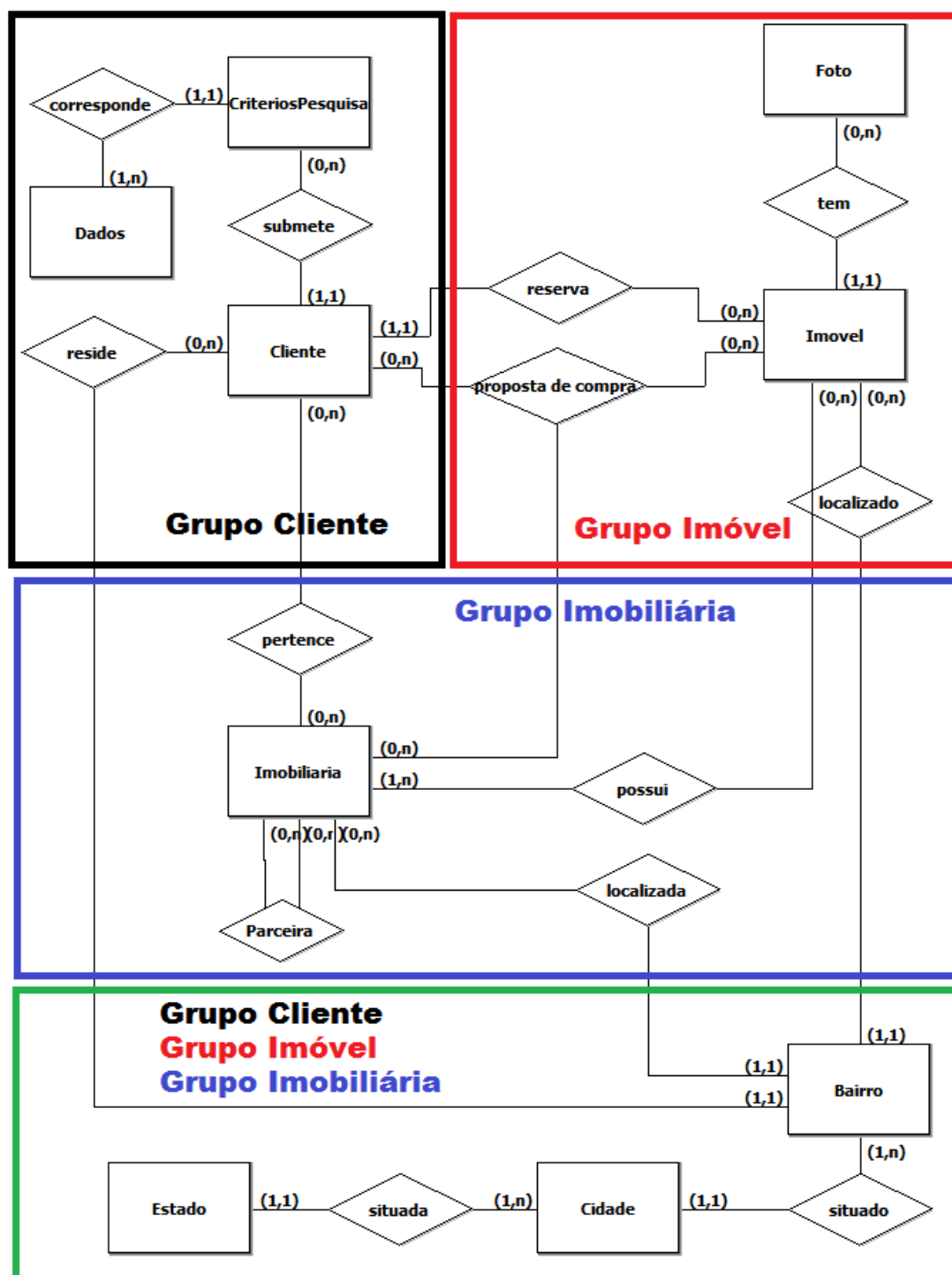


Figura 6 - Grupos de entidades

Identificados os grupos de entidades, é necessário listar em uma tabela os serviços candidatos, relacionando-os aos grupos de entidades que estes manipulam através de alguma operação CRUD (*Create, Retrive, Update e Delete*). Esta parte pode ser vista na Tabela 26.



**Tabela 26 - Associação de serviços candidatos com grupos de entidades**

Serviços candidatos	Grupos de Entidades		
	Cliente	Imobiliária	Imóvel
Registrar critérios de pesquisa	C		
Verificar abrangência de imobiliárias parceiras		R	
Listar imóveis da imobiliária OnLine			R
Listar imóveis de imobiliárias parceiras		R	R
Buscar imóveis		R	R
Invocar serviços de imobiliárias parceiras		R	
Consolidar resultado da pesquisa de imóveis		R	R
Reservar imóvel			C
Registrar proposta de compra	R	R	CR
Submeter proposta de compra	R	R	CR

O próximo passo é realizar o agrupamento dos serviços candidatos de acordo com o grau de semelhança, levando em consideração as entidades que eles manipulam. Este passo pode ser visto na Tabela 27. Nesta, a marcação 'H' é usada quando um grupo de serviços acessa uma entidade que é manipulada por outro grupo de serviços, sendo seu acesso a ela por meio da invocação dos serviços do grupo responsável por ela.

**Tabela 27 - Agrupamento de serviços candidatos de acordo com grupos de entidades manipuladas**

Serviços candidatos		Grupos de Entidades		
		Cliente	Imobiliária	Imóvel
G1	Registrar critérios de pesquisa	C		
G2	Invocar serviços de imobiliárias parceiras		R	
	Verificar abrangência de imobiliárias parceiras		R	
G3	Listar imóveis da imobiliária Online			R
	Listar imóveis de imobiliárias parceiras		R	R
	Reservar imóvel			C
	Buscar imóveis		R	R

	Consolidar resultado da pesquisa de imóveis		R	R
G4	Registrar proposta de compra	H (R)	R	H (CR)
	Submeter proposta de compra	H (R)	R	H (CR)

Após o agrupamento, pode-se decidir se os serviços de um determinado grupo serão implementados como um único serviço físico ou não.

#### 4.3.3.1 Seleção de serviços

Após a execução das etapas mostradas anteriormente, é possível decidir quais destes serviços devem ser projetados e implementados. Para auxiliar na seleção, Azevedo *et al.* (2011) propõem empregar as operações propostas por Marks e Bell *et al.* (2006) que podem ser realizadas para definir os serviços físicos, as quais são:

- Unificação de serviços: Identifica contexto de negócio e abstrações comuns para unir mais de um serviço, implementando-o como um serviço.
- Interseção de serviços: Identifica funcionalidades comuns em mais de um serviço, expondo-as como um novo serviço.
- Decomposição de serviços: Divide um serviço candidato que possua granularidade grossa em serviços mais simples.
- Subtração de serviços: Desconsidera funcionalidades em serviços candidatos que não serão implementadas fisicamente.
- Subconjunto de serviços: Agrega serviços em outro serviço já existente como serviço candidato.

Levando em consideração as informações obtidas nas etapas de identificação e análise, e utilizando a unificação de serviços proposta por Marks e Bell *et al.* (2006), o resultado da seleção de serviços pode ser observada na Tabela 28. Estes serviços serão projetados e implementados fisicamente.

**Tabela 28 - Seleção de serviços**

Serviços candidatos		Serviços selecionados
G1	Registrar critérios de pesquisa	Registrar critérios de pesquisa
G2	Invocar serviços de imobiliárias parceiras	Invocar serviços de imobiliárias parceiras
	Verificar abrangência de	Verificar abrangência de imobiliárias

	imobiliárias parceiras	parceiras
G3	Listar imóveis da imobiliária Online	Listar imóveis da imobiliária Online
	Listar imóveis de imobiliárias parceiras	Listar imóveis de imobiliárias parceiras
	Reservar imóvel	Reservar imóvel
	Buscar imóveis	Buscar imóveis
	Consolidar resultado da pesquisa de imóveis	Consolidar resultado da pesquisa de imóveis
G4	Registrar proposta de compra	Registrar proposta de compra
	Submeter proposta de compra	

## 4.4 Projeto

Esta seção apresenta o projeto dos serviços candidatos que serão implementados. A etapa de projeto se inicia com a separação dos serviços por tipo (dados ou lógica), como pode ser observado na Tabela 29.

**Tabela 29 - Identificação dos tipos de serviços**

Serviços candidatos	Tipo do serviço
Registrar critérios de pesquisa	Dados
Verificar abrangência de imobiliárias parceiras	Dados
Listar imóveis da imobiliária Online	Dados
Listar imóveis de imobiliárias parceiras	Dados
Buscar imóveis	Dados
Invocar serviços de imobiliárias parceiras	Dados
Consolidar resultado da pesquisa de imóveis	Dados
Reservar imóvel	Dados
Registrar proposta de compra	Dados

Esta separação deve ser feita, pois serviços de dados devem ser implementados de forma diferente dos serviços de lógica.

O segundo passo na etapa de projeto é a elaboração de modelos canônicos. O modelo canônico representa um conjunto de dados comum, que é independente de aplicações, mas que é compartilhado por elas. Segundo Woolf [2006], com o modelo canônico as aplicações não precisam negociar em formatos de mensagens entre si, podendo, simplesmente, utilizar o modelo de dados canônico existente. Os modelos canônicos elaborados neste trabalho se encontram no Apêndice 2.

Para o próximo passo, foi elaborado o caso de uso e o diagrama de sequência para o processo “Submeter proposta de compra”, que resultou no caso de uso “Pesquisar Imóvel” e no diagrama de sequência para o mesmo, apresentados no Apêndice 4. Este passo é importante, pois detalha operações e fluxos de operações, facilitando a implementação dos serviços.

## 4.5 Implementação

A partir da listagem dos serviços candidatos, apresentado na Tabela 12, os mesmos foram implementados na ferramenta NetBeans IDE e em seguida foram disponibilizados no Oracle Service Bus de acordo com o caso de uso apresentado no Apêndice 4. As descrições dos serviços implementados são apresentadas na Tabela 30.

De acordo com as necessidades de implementação, alguns serviços foram implementados na aplicação cliente, como os serviços *Buscar imóveis*, *Consolidar resultado de pesquisa de imóveis* e *Invocar serviços de imobiliárias parceiras*. Isto ocorreu porque a orquestração dos serviços foi realizada pela aplicação cliente, e não utilizando um padrão específico para isto, como, por exemplo, WS-BPEL.

O serviço *Registrar proposta de compra* foi considerado como um serviço de lógica na Tabela 30, uma vez que é verificado se o imóvel possui ou não uma proposta de compra aprovada. No caso de possuir uma proposta de compra aprovada deve retornar ao cliente que o imóvel já foi vendido, e caso contrario deve ser feito um registro da proposta de compra que corresponde à submissão da mesma para análise no processo “*Analisar proposta de compra*”. Este processo está fora do escopo deste trabalho. O serviço *Reservar imóvel* não foi implementado uma vez que durante discussões para desenvolvimento da aplicação identificamos que o imóvel só está reservado a um cliente se a proposta de compra for aceita pela imobiliária proprietária do imóvel. Logo, a regra de negócio “Reservar imóvel” deve ser revista na especificação do modelo de processo de negócio.

Com há o serviço *Listar imóveis de imobiliárias parceiras* e este trabalho tem o objetivo de demonstrar e enfatizar as funcionalidades que SOA permite, como por exemplo, a integração de sistemas distribuídos, foram consideradas duas imobiliárias

parceiras, com o nome de Imobiliária Inglesa e Imobiliária Alemã. Os modelos canônicos destas imobiliárias também foram elaborados e mais detalhes são apresentados no Apêndice 2. Assim sendo, o serviço *Listar imóveis de imobiliárias parceiras* foi implementado para cada imobiliária parceira, resultando nos serviços: *Listar imóveis de imobiliária alemã* e *Listar imóveis de imobiliária inglesa*.

Os serviços *Listar imóveis da imobiliária Online*, *Listar imóveis de imobiliária alemã* e *Listar imóveis de imobiliária inglesa*, possuem a mesma saída que são todos os dados referentes ao imóvel e o CNPJ da imobiliária.

Já os serviços *Buscar imóveis*, *Consolidar resultado de pesquisa de imóveis* e *Invocar serviços de imobiliárias parceiras*, implementados na aplicação cliente, e consolidados na Tabela 30 como o serviço *Consolidar busca*, tem como objetivo invocar os serviços *Listar imóveis da imobiliária Online*, *Listar imóveis de imobiliária alemã* e *Listar imóveis de imobiliária inglesa* de acordo com as imobiliárias que atendem os bairros pesquisados e consolidar em uma lista única o retorno dos mesmos.

Outra observação importante é que sempre que uma proposta de compra é registrada (serviço *Registrar proposta de compra*) para um imóvel de uma imobiliária parceira os dados referentes ao imóvel em questão são armazenados junto ao banco de dados da imobiliária Online, uma vez que é necessário o registro dos dados do imóvel para referência, e caso o imóvel já possua uma proposta de compra “Aprovada” o registro da proposta é rejeitado.

A partir destas observações a Tabela 30 e os diagramas de classes apresentados no Apêndice 5 foram elaborados com objetivo de explicitar em detalhes cada serviço implementado e sua origem.

**Tabela 30 - Serviços implementados**

Nome do serviço implementado	Verificar abrangência da imobiliária alemã	Verificar abrangência da imobiliária inglesa	Verificar abrangência da imobiliária online	Consolidar abrangência	Registrar critérios de pesquisa	Listar imóveis da imobiliária alemã	Listar imóveis da imobiliária inglesa	Listar imóveis da imobiliária online	Consolidar busca	Registrar proposta de compra
Serviço candidato de origem	- Verificar abrangência de imobiliárias parceiras				- Registrar critérios de pesquisa	- Listar imóveis de imobiliárias parceiras.		- Listar imóveis da imobiliária Online	- Buscar imóveis; - Consolidar resultado da pesquisa de imóveis; - Invocar serviços de imobiliárias parceiras.	- Registrar proposta de compra
Tipo	Dados									Lógica
Entrada				Lista de bairros	- CPF; - Lista de bairros; - Valor mínimo; - Valor máximo; - Número de quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.	- Lista de bairros; - Valor mínimo; - Valor máximo; - Número de quartos; - Se possui vaga de garagem; - Se é sol da manhã e/ou sol da tarde.			- CNPJ da imobiliária; - ID do imóvel; - Preço; - Área; - Número de quartos; - Se possui vaga na garagem; - Se é sol da manhã e/ou sol da tarde; - Logradouro; - Número; - Complemento; - Bairro; - Cidade; - Estado; - Foto; - Outros detalhes do imóvel.	- Nome do cliente; - CPF; - CNPJ; - ID de origem do imóvel; - Valor; - Descrição.
Saída	Lista de bairros.					- CNPJ da imobiliária; - ID do imóvel; - Preço; - Área;				Proposta de compra registrada ou não.

						<ul style="list-style-type: none"> <li>- Número de quartos;</li> <li>- Se possui vaga na garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- Logradouro;</li> <li>- Número;</li> <li>- Complemento;</li> <li>- Bairro;</li> <li>- Cidade;</li> <li>- Estado;</li> <li>- Foto;</li> <li>- Outros detalhes do imóvel.</li> </ul>				
Descrição	Tem como objetivos retornar os bairros que a imobiliária alemã abrange e que possui imóveis disponíveis para venda através da imobiliária Online.	Tem como objetivos retornar os bairros que a imobiliária inglesa abrange e que possui imóveis disponíveis para venda através da imobiliária Online.	Tem como objetivo retornar os bairros que a Imobiliária Online possui imóveis.	Tem como objetivo consolidar em uma lista única o resultado retornado pelos serviços:	O sistema imobiliária online deve registrar os seguintes critérios pesquisados:	É feita uma busca por imóveis na base de dados da imobiliária alemã, com o objetivo de retornar os imóveis que a mesma possui disponível para venda a partir do sistema da imobiliária online de acordo com os critérios de pesquisa informados.	É feita uma busca por imóveis na base de dados da imobiliária inglesa, com o objetivo de retornar os imóveis que a mesma possui disponível para venda a partir do sistema da imobiliária Online de acordo com os critérios de pesquisa informados.	É feita uma busca por imóveis na base de dados da imobiliária Online, de acordo com os critérios de pesquisa informados.	Tem como objetivo consolidar em uma lista única o resultado retornado pelos serviços:	O sistema imobiliária Online deve verificar se o imóvel não possui nenhuma proposta de compra aprovada, se possuir uma proposta de compra aprovada dever retornar ao cliente que o imóvel já foi vendido e caso contrario registrar uma proposta com os seguintes critérios:
				<ul style="list-style-type: none"> <li>- Verificar abrangência da imobiliária alemã;</li> <li>- Verificar abrangência da imobiliária inglesa;</li> <li>- Verificar abrangência da imobiliária Online.</li> </ul>	<ul style="list-style-type: none"> <li>- CPF do cliente</li> <li>- Data da pesquisa</li> <li>- Lista com o nome dos bairros pesquisados e CNPJ da imobiliária ao qual o bairro pertence;</li> <li>- Nome dos bairros pesquisados;</li> <li>- Valor mínimo</li> <li>- Valor máximo</li> <li>- Número de quartos;</li> <li>- Se possui vaga de garagem;</li> <li>- Se é sol da manhã e/ou</li> </ul>				<ul style="list-style-type: none"> <li>- Listar imóveis da imobiliária alemã;</li> <li>- Listar imóveis da imobiliária inglesa;</li> <li>- Listar imóveis da imobiliária Online.</li> </ul>	<ul style="list-style-type: none"> <li>- CPF;</li> <li>- Valor;</li> <li>- Data da proposta;</li> <li>- CNPJ;</li> <li>- ID de origem;</li> <li>- ID do imóvel;</li> <li>- Valor;</li> <li>- Descrição</li> </ul> <p>Além disso, caso o imóvel</p>

					sol da tarde.					em questão seja de uma imobiliária parceira os dados referentes ao imóvel devem ser registrados junto à imobiliária Online.
Observações				Deve-se ressaltar que a funcionalidade referente a este serviço foi, na realidade, implementada na aplicação cliente.					Deve-se ressaltar que a funcionalidade referente a este serviço foi, na realidade, implementada na aplicação cliente.	



## CAPÍTULO 5: DISPONIBILIZAÇÃO NO ESB DOS SERVIÇOS IMPLEMENTADOS

---

Este capítulo tem como objetivo apresentar a disponibilização dos serviços implementados na seção 4.5 em um ESB. Para esta etapa foi utilizada a ferramenta Oracle Service Bus versão 11gR1, que é uma extensão do Oracle SOA Suite 11g.

### 5.1 Executando o servidor de aplicação do ESB

Para realizar as etapas a seguir é necessário iniciar o servidor de aplicação do domínio, criado de acordo com o passo a passo apresentado no Apêndice 6. Para iniciar o servidor é necessário acessar o local do domínio que foi especificado, e executar o arquivo *startWebLogic.cmd*. Depois que o servidor tiver em execução deve-se acessar o endereço *http://localhost:7001/sbconsole*, informando o nome de usuário e senha que foi definido na criação do domínio.

### 5.2 Cadastrando o serviço

Nesta seção iremos apresentar o cadastramento de um serviço desenvolvido no Netbeans. O serviço mostrado será o serviço *Registrar Critérios de Pesquisa*, pois os passos abaixo são genéricos para a disponibilização de todos os serviços a partir de seus WSDL (*Web Service Definition Language*).

O primeiro passo é criar uma pasta para o serviço e três subpastas, onde ficarão armazenados seus recursos (WSDL, Serviço de proxy e Serviço de negócio) como mostrado na Figura 7.

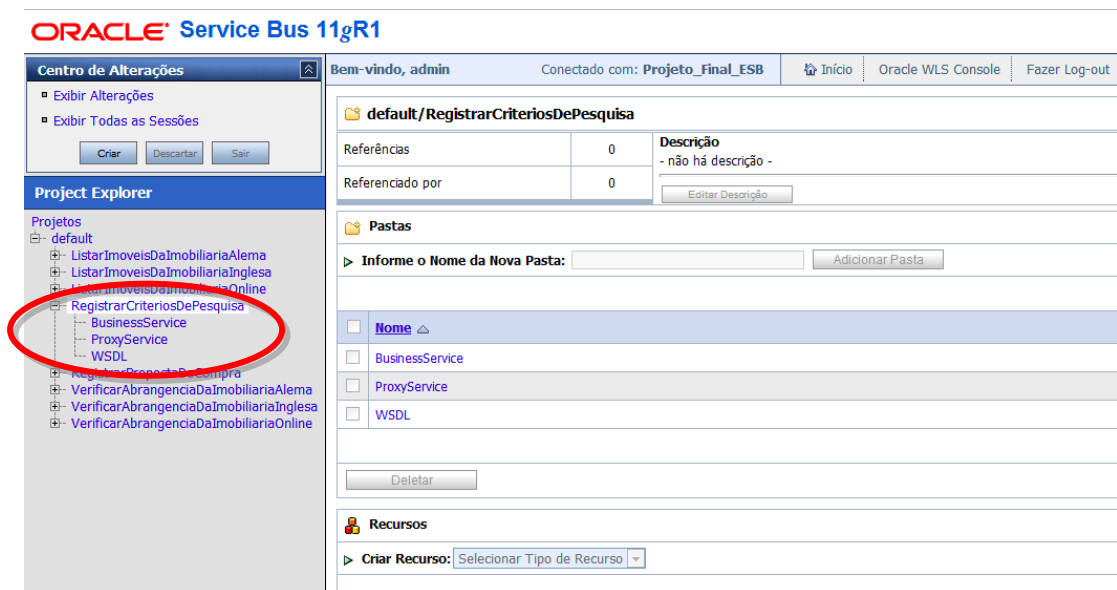


Figura 7 - Criação das pastas

### 5.2.1 Importação do WSDL

Nesta subseção mostraremos como importar o WSDL, que é uma linguagem utilizada para descrever serviços de negócio funcionando como um contrato do serviço.

Dentro da subpasta *WSDL*, importa-se o WSDL do serviço através de sua URL ou através de um arquivo (.jar). Neste caso, a importação foi feita através de *Recursos do URL*, como mostrado na Figura 8.

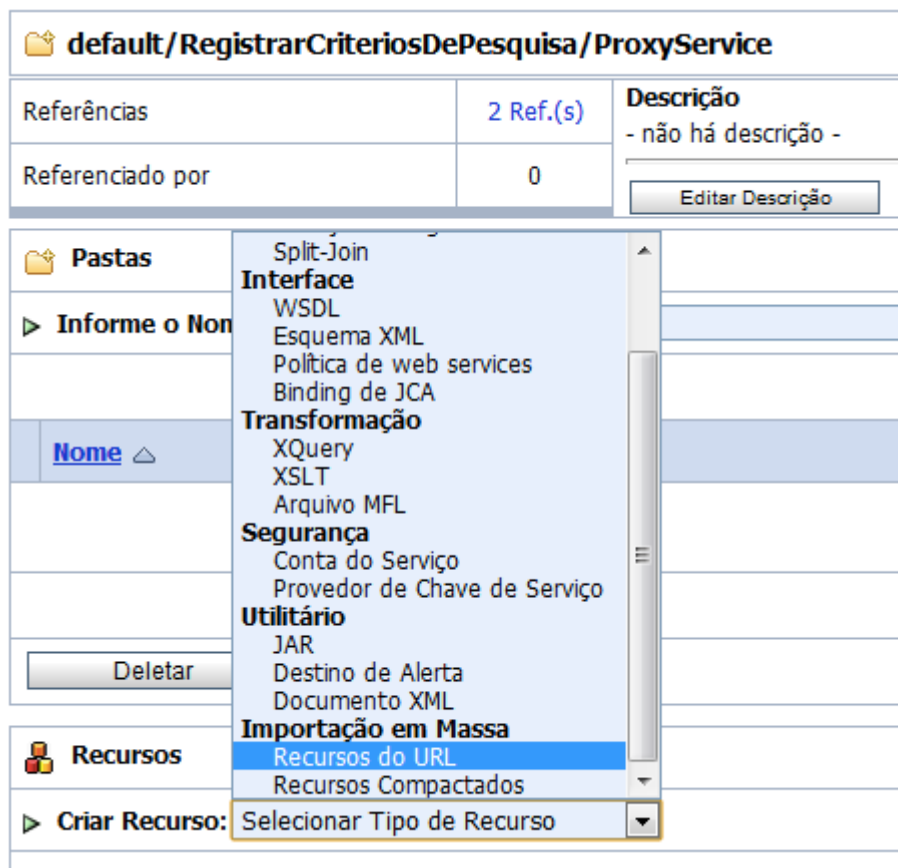


Figura 8 - Selecionando o recurso WSDL

Deve-se, então, preencher o campo destinado à *URL/Caminho* com o endereço do WSDL implementado no Netbeans, além do *Nome do Recurso* e *Tipo de recurso*, como mostrado na Figura 9.

Carregar Recursos	
Carregar Recursos do URL	
URL/Caminho*	<input type="text" value="http://localhost:8080/ProjetoFinal/RegistrarCriteriosDePesquisa?WSDL"/>
Nome do Recurso*	<input type="text" value="RegistrarCriteriosDePesquisaWSDL"/>
Tipo de Recurso*	<input type="text" value="WSDL"/>
<input type="button" value="Próximo &gt;&gt;"/> <input type="button" value="Cancelar"/>	

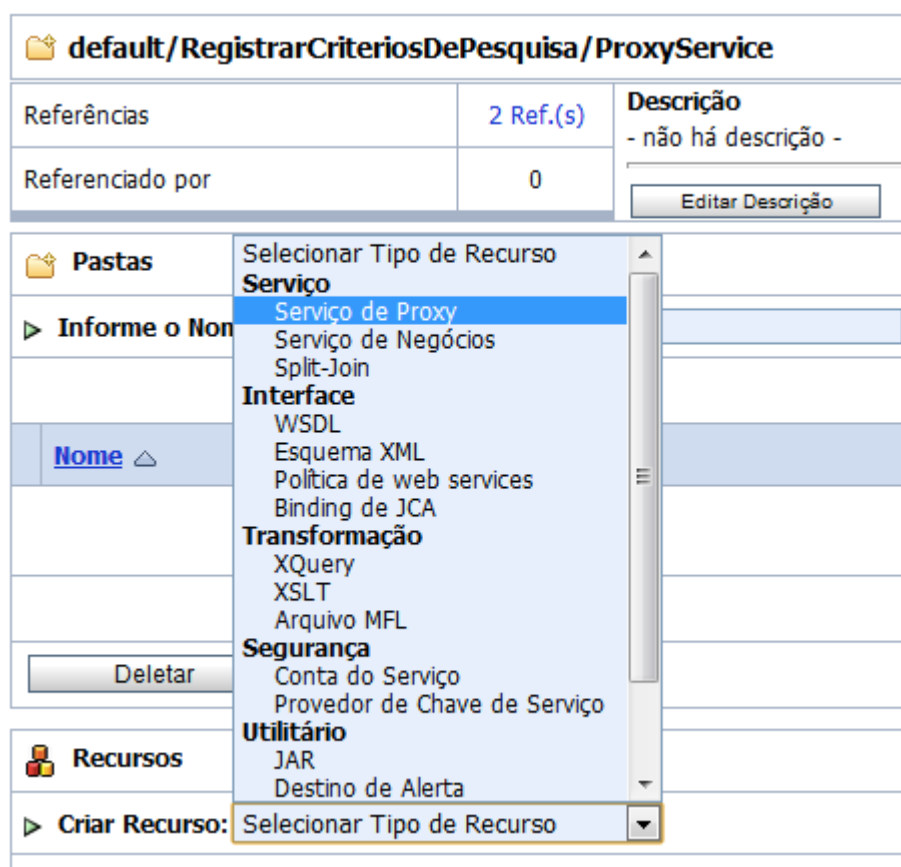
Figura 9 - Definindo os campos do WSDL

### 5.2.2 Implementação do Proxy Service

Nesta subseção mostraremos como definir o recurso de proxy. O serviço de proxy é o serviço do ESB que recebe a mensagem de solicitação do consumidor e

repassa para o serviço real, o qual é executado no servidor de aplicação. Além do roteamento, o serviço de proxy pode realizar outras tarefas tais como realizar transformações necessárias para converter a estrutura da mensagem enviada pelo consumidor para a estrutura esperada pelo provedor do serviço e vice-versa, tratar questões de segurança, validar estrutura das mensagens etc.

Dentro da subpasta *ProxyService*, deve-se criar o recurso *Serviço de Proxy*, como mostrado na Figura 10.



**Figura 10 - Selecionando o recurso Proxy Service**

Deve-se então, preencher os campos *Nome do serviço* e *Tipo de Serviço* como mostrado na Figura 11. No campo *Tipo de Serviço*, dentre as opções possíveis, foi selecionada a opção *Web Service de WSDL* e foi definido o WSDL importado na subseção 5.2.1.

Criar um Serviço de Proxy (default/RegistrarCriteriosDePesquisa/ProxyService/)	
Configuração Geral	
Nome do Serviço*	RegistrarCriteriosDePesquisaProxy
Descrição	
Tipo de Serviço*	<div> <b>Criar um Novo Serviço</b> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> Web Service de WSDL <div> default/RegistrarCriteriosDePesquisa/WSI Procurar... RegistrarCriteriosDePesquisaPort (porta) </div> </li> <li><input type="radio"/> Serviço do Tipo de Transporte</li> <li><input type="radio"/> Serviço de Mensagem</li> <li><input type="radio"/> Qualquer Serviço de SOAP <div> SOAP 1.1 </div> </li> <li><input type="radio"/> Qualquer Serviço XML</li> </ul> </div> <div> <b>Criar do Serviço Existente</b> <ul style="list-style-type: none"> <li><input type="radio"/> Serviço de Negócios <div> Procurar... </div> </li> <li><input type="radio"/> Serviço de Proxy <div> Procurar... </div> </li> </ul> </div>
<div> Próximo &gt;&gt; Último &gt;&gt; Cancelar </div>	

**Figura 11 - Definindo os campos do Proxy Service**

### 5.2.3 Implementação do Business Service

Nesta subseção mostraremos como o serviço de negócio. O serviço de negócio corresponde ao serviço que está executando no provedor, ou seja, ele não executa no ESB. O ESB faz referência ao serviço de negócio.

Dentro da subpasta *BusinessService*, deve-se criar o recurso Serviço de Negócio, como exibido na Figura 12.

default/RegistrarCriteriosDePesquisa/BusinessService		
Referências	1 Ref.(s)	<b>Descrição</b> - não há descrição - <input type="button" value="Editar Descrição"/>
Referenciado por	1 Ref.(s)	
<div> <div> <b>Pastas</b>            Informe o Nome  <input type="text"/>  <input type="button" value="Deletar"/> </div> <div> <b>Recursos</b>            Criar Recurso:           <div>             Selecionar Tipo de Recurso              Serviço                Serviço de Proxy                <b>Serviço de Negócios</b>                Split-Join              Interface                WSDL                Esquema XML                Política de web services                Binding de JCA              Transformação                XQuery                XSLT                Arquivo MFL              Segurança                Conta do Serviço                Provedor de Chave de Serviço              Utilitário                JAR                Destino de Alerta           </div> </div> </div>		

**Figura 12 - Selecionando o recurso Business Service**

Deve-se então, preencher os campos *Nome do serviço* e *Tipo de Serviço* como mostrado na Figura 13. No campo *Tipo de Serviço*, dentre as opções possíveis, foi selecionada a opção *Web Service de WSDL* e foi definido o WSDL importado na subseção 5.2.1. Vale ressaltar que o serviço de proxy faz o roteamento para o serviço de negócio. O WSDL é utilizado para realizar a configuração da interface destes serviços no ESB.

**Criar um Serviço de Negócios (default/RegistrarCriteriosDePesquisa/BusinessService/)**

**Configuração Geral**

<b>Nome do Serviço*</b>	RegistrarCriteriosDePesquisaService		
<b>Descrição</b>			
<b>Tipo de Serviço*</b>	<div> <b>Criar um Novo Serviço</b> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> Web Service de WSDL           <div>             default/RegistrarCriteriosDePesquisa/WSI              RegistrarCriteriosDePesquisaPort (porta)           </div> </li> <li><input type="radio"/> Serviço do Tipo de Transporte</li> <li><input type="radio"/> Serviço de Mensagem</li> <li><input type="radio"/> Qualquer Serviço de SOAP           <div>SOAP 1.1 ▼</div> </li> <li><input type="radio"/> Qualquer Serviço XML</li> </ul> </div> <div> <b>Criar do Serviço Existente</b> <ul style="list-style-type: none"> <li><input type="radio"/> Serviço de Negócios           <div></div> </li> <li><input type="radio"/> Serviço de Proxy           <div></div> </li> </ul> </div>		

**Figura 13 - Definindo os campos do Business Service**

## 5.2.4 Configuração do Serviço de proxy

Nesta subseção mostraremos como configurar o serviço de proxy, definindo o fluxo das mensagens, cujo primeiro passo (criação de rota para serviço) é apresentado na Figura 14.



**Figura 14 - Definindo o fluxo de mensagens do proxy**

Nesta configuração, foi editada o *RouteNode1* e especificado para qual serviço a mensagem vai ser roteada, indicando qual função deste serviço será chamada. Esta configuração pode ser observada na Figura 15, onde o serviço de proxy *RegistrarCriteriosDePesquisaProxy* é roteado para o serviço de negócio *RegistrarCriteriosDePesquisaService*, definido na seção 5.2.3, chamando a função *registrarPesquisa*.





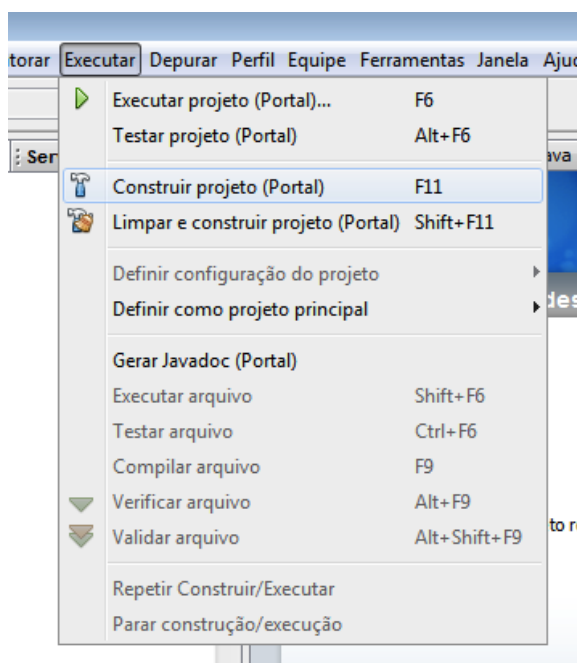
## CAPÍTULO 6: APLICAÇÃO CLIENTE

---

Este capítulo tem como objetivo apresentar a aplicação cliente desenvolvida na ferramenta NetBeans IDE 7.0.1 e como implantar essa aplicação web no servidor de aplicação Glassfish, fazendo com que a aplicação possa ser executada independente do NetBeans. A aplicação cliente consome os serviços disponibilizados no ESB e exportados a partir de seus WSDL, como apresentado na seção 5.3.

### 6.1 Implantando a aplicação cliente e serviços no Glassfish

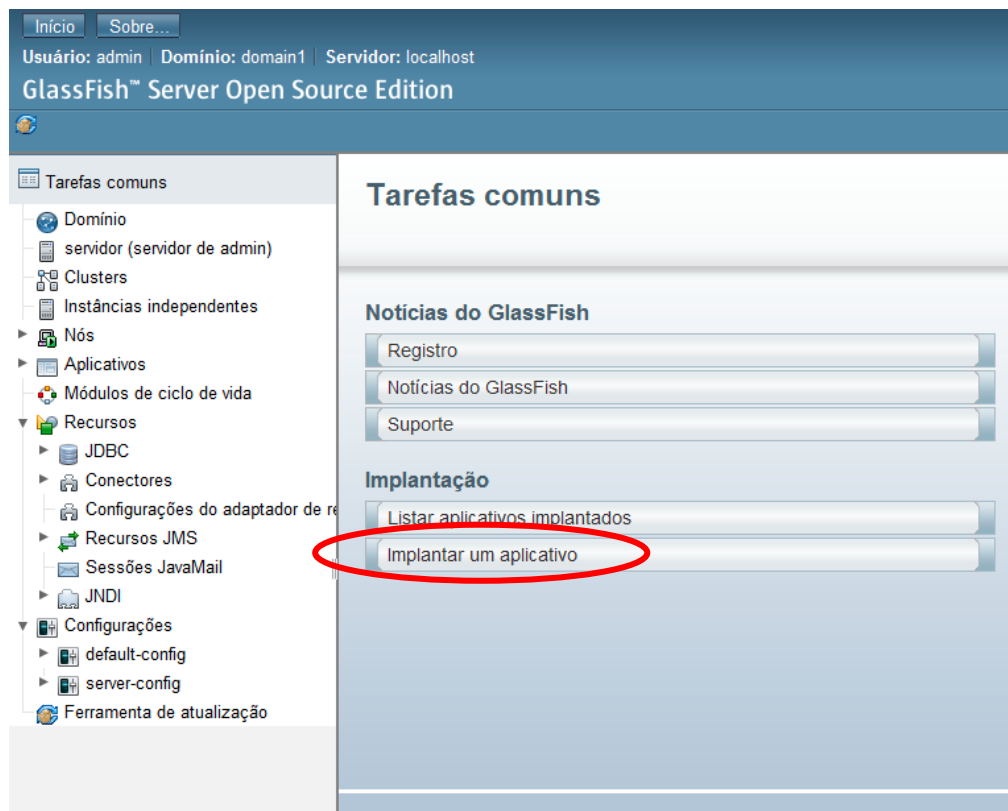
Para implantar a aplicação cliente e os serviços no Glassfish é necessário exportá-los como arquivos *.war* e instalá-los no Glassfish da seguinte forma. Inicie o NetBeans, abra o projeto e, na barra de fermentas, acesse *Executar* e clique em *Construir projeto*. Este processo é mostrado na Figura 17.



**Figura 17 - Construindo projeto no NetBeans**

Em seguida execute o arquivo *startserv.bar* que se encontra na pasta *Program Files\glassfish-3.1.1\glassfish\bin\*. Feito isso acesse o endereço

`http://localhost:4848`, no navegador, e selecione a opção Implantar um aplicativo, como apresentado na Figura 18.



**Figura 18 - Console do Glassfish**

Escolha o arquivo `.war` exportado pelo Netbeans e clique no botão *OK*, como apresentado na Figura 19. Assim temos o serviço implantado no Glassfish.

#### Implantar aplicativos ou módulos

Especifique a localização do aplicativo ou módulo a ser implantado. O aplicativo pode ser empacotado em um arquivo ou especificado como um diretório.

Localização: ☒ Pacote do arquivo a ser enviado ao servidor

Nenhum arquivo selecionado

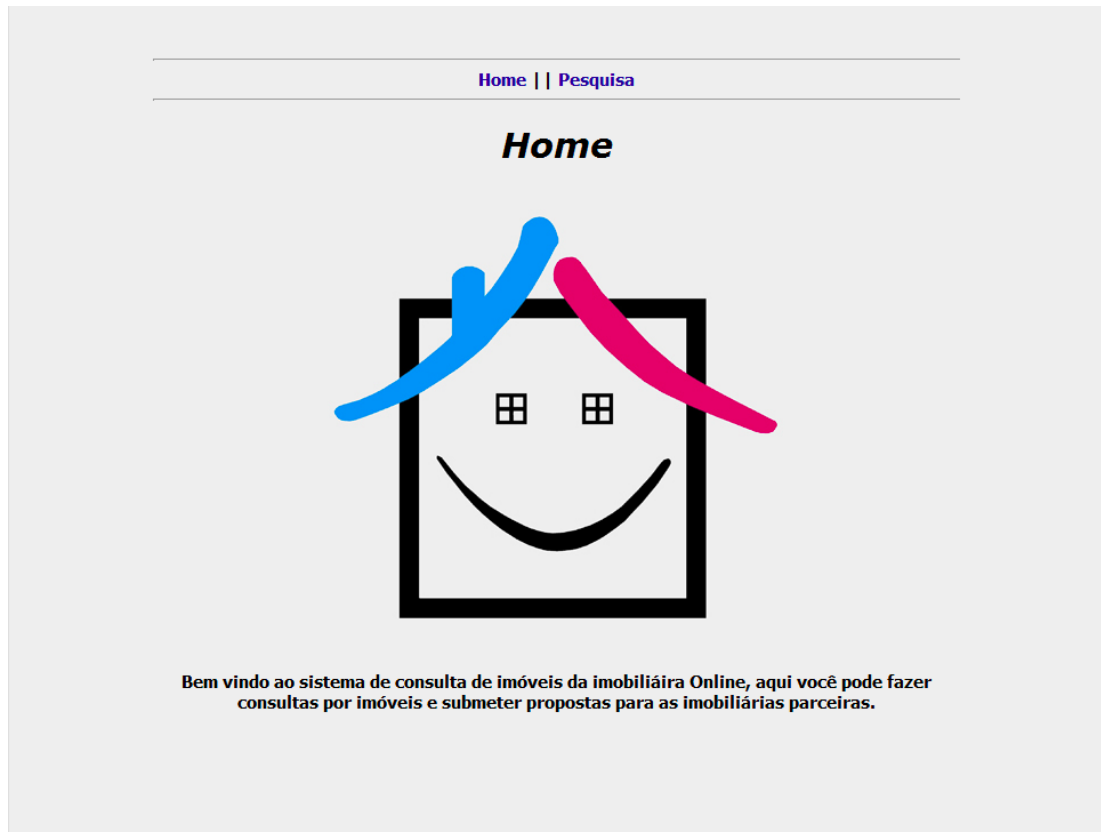
☐ Diretório ou arquivo local empacotado que pode ser acessado a partir do servidor GlassFish

Tipo: \*  ▼

**Figura 19 - Importando .war**

## 6.2 Aplicação cliente dos serviços

A aplicação cliente tem como objetivo ser o portal onde os clientes da imobiliária Online irão pesquisar seus imóveis e imóveis de imobiliárias parceiras. Logo que o cliente acessa o portal (Figura 20) lhe é apresentado uma página inicial com a opção de pesquisa.



**Figura 20 - Página inicial do portal**

Quando o cliente seleciona a opção pesquisa, lhe são apresentados os campos para pesquisar pelos imóveis disponíveis (Figura 21), como especificado no caso de uso apresentado no Apêndice 4.

[Home](#) | [Pesquisa](#)

## Pesquisar imóvel

Informe o seu CPF:

**Dados para pesquisa de imóvel:**

Selecione o(s) bairro(s): 

Ipanema

Tijuca

Barra da Tijuca

Copacabana

Digite o valor mínimo:

Digite o valor máximo:

Digite o número de quartos:

Possui vaga de garagem: ☐ Sim ☐ Não

Horário do sol: ☐ Manhã ☐ Tarde

**Figura 21 - Página de pesquisa do portal**

Após preencher os campos de pesquisa, onde todos os campos são obrigatórios, exceto o campo de valor máximo ou valor mínimo, no qual no mínimo um dos campos deve estar preenchido, são apresentado os imóveis disponíveis como exemplificado na Figura 22.

[Home](#) | [Pesquisa](#)

### Dados dos imóveis

Imóvel(s) encontrados:

Imobiliária	Preço	Área	Número de quartos	Possui vaga de garagem	Horário do sol	Logradouro	Número	Complemento	Bairro	Cidade	Estado	Opções
Imobiliária Rio de Janeiro	1800000.0	150m2	2	Sim	Manha	Av Vieira Souto	396	AP101	Ipanema	Rio de Janeiro	Rio de Janeiro	<input type="button" value="Detalhes"/> <input type="button" value="Submeter proposta"/>
Imobiliária Rio de Janeiro	1500000.0	180m2	2	Sim	Manha	Av Atlantica	800	AP104	Copacabana	Rio de Janeiro	Rio de Janeiro	<input type="button" value="Detalhes"/> <input type="button" value="Submeter proposta"/>
Imobiliária Rio de Janeiro	650000.0	110m2	2	Sim	Manha	Rua Haddock Lobo	400	AP707	Tijuca	Rio de Janeiro	Rio de Janeiro	<input type="button" value="Detalhes"/> <input type="button" value="Submeter proposta"/>
Imobiliária Londres	1200000.0	250m2	2	Sim	Manha	Rua Sousa	366	BL1 AP1402	Jardim Paraventi	Guarulhos	Sao Paulo	<input type="button" value="Detalhes"/> <input type="button" value="Submeter proposta"/>
Imobiliária Berlim	950000.0	250m2	2	Sim	Manha	Rua Dep Moreira Rocha	1903	AP402	Meireles	Fortaleza	Ceara	<input type="button" value="Detalhes"/> <input type="button" value="Submeter proposta"/>
Imobiliária Berlim	2000000.0	190m2	2	Sim	Manha	Av Beira Mar	1387	AP1203	Praia de Iracema	Fortaleza	Ceara	<input type="button" value="Detalhes"/> <input type="button" value="Submeter proposta"/>

**Figura 22 - Lista de imóveis encontrados**

A partir da lista de imóveis disponíveis, o cliente pode clicar na opção para exibir mais detalhes do imóvel, no qual são exibidos todos os detalhes referentes ao imóvel e a imobiliária no qual o mesmo pertence (Figura 23), ou então o cliente pode

selecionar a opção submeter proposta, na qual ele é redirecionado imediatamente para uma página de submissão de propostas (Figura 24).

Dados do imóvel											
Área	Preço	Número de quartos	Possui vaga de garagem	Horário do sol	Outros detalhes	Logradouro	Número	Complemento	Bairro	Cidade	Estado
150m2	1800000.0	2	Sim	Manhã	null	Av Vieira Souto	396	AP101	Ipanema	Rio de Janeiro	Rio de Janeiro

Dados da imobiliária										
CNPJ	Nome	Telefone 1	Telefone 2	Telefone 3	Logradouro	Número	Complemento	Bairro	Cidade	Estado
14.218.935/0001-27	Imobiliária Rio de Janeiro	(21) 2440-7766	null	null	Av Nossa Senhora de Copacabana	900	Loja 2	Copacabana	Rio de Janeiro	Rio de Janeiro

[Voltar](#) [Submeter proposta](#)

Figura 23 - Página de detalhes do imóvel

Na página de detalhes do imóvel o cliente pode selecionar a opção submeter proposta para o imóvel em questão, no qual ele é direcionado para a página de submissão de propostas, representado pela Figura 24, ou então, clicar na opção voltar, que volta para a lista de imóveis disponíveis.

**Submeter proposta**

Nome:

CPF:

Valor para proposta:

Descrição:

[Submeter](#)

Figura 24 - Página de submissão de propostas

Após preencher todos os campos da página de proposta e clicar na opção submeter proposta, o cliente confirma os dados da proposta e lhe é exibido uma página informando se a proposta foi cadastrada com sucesso ou não (Figura 25).

## ***Detalhes do cadastro da proposta***

Proposta cadastrada com sucesso!

**Figura 25 - Página com detalhes do cadastro da proposta**

## CAPÍTULO 7: CONCLUSÃO

---

Neste trabalho foi realizada a aplicação prática dos métodos para identificação, análise, projeto e implementação de serviços propostos por Azevedo *et al.*, (2009a, 2009b, 2011) e Dirr *et al.* (2012), além de realizar o cadastramento dos serviços implementados em um ESB e desenvolver uma aplicação web para consumir os serviços disponibilizados no ESB. O ESB utilizado foi o Oracle Service Bus, uma extensão para o Oracle SOA Suite. Esta escolha foi baseada após uma análise das ferramentas disponíveis, onde se concluiu que o Oracle Service Bus seria interessante devido à sua relevância no mercado, avaliação de ferramentas ESB e ao grande volume de documentação e material disponíveis.

Desta forma, verificou-se a aplicabilidade dos métodos de identificação, análise, projeto e implementação, assim como as vantagens da utilização de um ESB para um ambiente SOA, pois aplicações desenvolvidas em ambientes diferentes podem ser disponibilizadas para clientes através de aplicações que façam o uso dos serviços disponibilizados pelo ESB.

A utilização das heurísticas definidas por Azevedo *et al.* (2009a, 2009b e 2011) e Diirr *et al.* (2012) foi de grande importância para o desenvolvimento dos serviços, pois auxiliaram como um guia em cada etapa de desenvolvimento, além de proporcionar, entre outras vantagens, uma maior facilidade de reuso e um mapeamento alinhado ao negócio. As etapas de identificação, análise e projeto de serviços foram facilitadas pelas explicações e exemplos descritos nos referidos trabalhos, mas demandaram um tempo maior devido ao seu nível de detalhes. A etapa de projeto também demandou tempo, uma vez que necessita de uma dedicação maior para definir os modelos canônicos e físicos, além dos diagramas de sequência. A etapa de implementação não foi tão custosa devido ao nível de detalhes das informações geradas pelas etapas anteriores, o que foi de fato muito útil para agilizar esta etapa. Desta forma, verificou-se que o volume de informações produzidas pelas etapas iniciais do método, apesar de demandarem mais tempo, agilizou a etapa de implementação dos serviços. Isto se deve ao nível de detalhe das informações obtidas pela aplicação do método.

Durante a aplicação do método, notou-se a carência de informações detalhadas no modelo de processo de negócio utilizado. Desta forma, foram realizados ajustes no modelo, complementando com maiores informações. Com isto, verificou-se a importância e a necessidade da utilização de processos bem detalhados para que se obtenham bons resultados a partir da aplicação do método estudado.

A utilização de um ESB foi de grande importância para a integração entre as imobiliárias no cenário utilizado, tornando-se uma ferramenta de grande relevância para um projeto com sistemas distribuídos. Verificou-se que o ESB diminui o acoplamento, permitindo que as aplicações clientes façam referência direta ao ESB, sem necessidade de conhecer a localização dos serviços, o que traz diversas possibilidades como, por exemplo, a rápida integração entre possíveis futuras imobiliárias parceiras.

Quanto ao processo de cadastramento e disponibilização dos serviços no Oracle Service Bus, podemos destacar que esta é uma etapa menos custosa se o desenvolvedor já possui conhecimento prévio da ferramenta, caso contrário, será necessário o estudo dos diversos manuais e tutoriais disponíveis tanto no site da ferramenta, quanto em livros e documentos.

Como trabalhos futuros propomos:

- Elaboração de heurísticas para sistematizar as propostas de Marks e Bell (2006) para as operações de unificação de serviços, interseção de serviços, decomposição de serviços, subtração de serviços e subconjunto de serviços.
- Elaboração de heurísticas para explicitar como o uso de ESB deve ser feito nas fases de projeto e implementação de serviços.
- Elaboração de heurísticas para integração de dados, uma vez que é frequente existir em aplicações distribuídas necessidade de integração de dados com estrutura e semânticas diferentes.
- Aplicação dos métodos nos processos “Analisar proposta de compra” e “Efetivar proposta de compra” do macro processo “Comprar imóvel”;



- Extensão da implementação do portal web, com o cadastro para novos clientes e novas funcionalidades elicítadas durante a aplicação dos métodos nos processos subsequentes.

## CAPÍTULO 8: REFERÊNCIAS BIBLIOGRÁFICAS

---

AZEVEDO, L.G.; BAIÃO, F.; SANTORO, F.; SOUZA, J. **A Business Aware Service Identification and Analysis Approach**. In: International Conference Information Systems (IADIS), 2011, Avila, Spain.

AZEVEDO, L.; PEREIRA, V.; REVOREDO, K; SOUZA, J.; SANTORO, F.; BAIÃO, F.; SOUSA, H. P. **Metodologia de identificação de serviços a partir da modelagem de processos de negócio**. Relatórios Técnicos do DIA/UNIRIO (RelaTe-DIA), RT-0021/2009, 2009b. Disponível (também) em: <http://seer.unirio.br/index.php/monografiasppgi>.

AZEVEDO, L.; SANTORO, F.; BAIÃO, F.; FURTADO, C; PEREIRA, V. **Arquitetura Orientada a Serviços**. Relatórios Técnicos do DIA/UNIRIO (RelaTe-DIA), RT-0023/2009, 2009a. Disponível (também) em: <http://seer.unirio.br/index.php/monografiasppgi>.

AZEVEDO, L. G.; SANTORO, F.; BAIÃO, F.; SOUZA, J. F.; REVOREDO; PEREIRA; HERLAIN. **A Method for Service Identification from Business Process Models in a SOA Approach**. In: 10th Workshop on Business Process Modeling, Development, and Support (BMDS'09), 2009, Amsterdam. Enterprise, Business-Process and Information Systems Modeling. Berlin Heidelberg : Springer, 2009. v. 29. p. 99-112.

BEZERRA, E. **Princípios de análise e projeto de sistemas com UML**. Campus, 2006, 308 p.

DIIRR, T.; AZEVEDO, L. G.; SANTORO, F.; BAIÃO, F.; FÁRIA, F. **Practical Approach for Service Design and Implementation**. IADIS International Conference in Information Systems, Berlim, Alemanha, 2012.

ERICKSON, J.; SIAU, K. **Web Services, Service-Oriented Computing, and Service-Oriented Architecture: Separating Hype from Reality**. Journal of Database Management, 19(3): 42-54, 2008.

ERL, T. **Service-Oriented Architecture: concepts, technology, and design**. Prentice Hall, 2005.

ERL, T. **SOA Principles of Service Design**, 1º ed., Prentice Hall, 2007.

GILPIN, M. 2010. **From The Field: The First Annual Canonical Model Management Forum**. Forrester Blogs. [http://blogs.forrester.com/print/mike\\_gilpin/10-03-15-field\\_first\\_annual\\_canonical\\_model\\_management\\_forum](http://blogs.forrester.com/print/mike_gilpin/10-03-15-field_first_annual_canonical_model_management_forum).

GU, Q., LAGO, P. **A Stakeholder-Driven Service Life Cycle Model for SOA**. In: 2nd International Workshop on Service Oriented Software Engineering: in Conjunction with the 6th ESEC/FSE Joint Meeting, pp. 1–7, 2007.

HEWITT, E. **Java SOA Cookbook**. O'Reilly, 2009.

JOSUTTIS, N. **SOA in practice: The Art of Distributed System Design**. O'Reilly, 2007.

KOHLBORN, T.; KORTHAUS, A.; CHAN, T.; ROSEMAN, M. **Identification and Analysis of Business and Software Services - A Consolidated Approach**, IEEE Transactions on Services Computing, 2009.

MARKS, E. A.; BELL, M. **Service-Oriented Architecture: a planning and implementation guide for business and technology**, John Willey & Sons Inc, 2006.

PAPAZOGLU, MIKE P.; HEUVEL, WILLEM-JAN. **Service oriented architectures: approaches, technologies and research issues**, VLDB Journal, Springer-Verlag, 2007.

VOLLMER, K., GILPIN, M., ROSE, S. **The Forrester Wave™: Enterprise service Bus, Q2 2011**. The Forrester Research, Abril 2011.








WOOLF, B. **Introduction to SOA governance**, 2006. Disponível em: <<http://www.ibm.com/developerworks/library/ar-servgov>>. Acessado em 10 Dez. 2010.







## APÊNDICE 1 – PROCESSO SUBMETER PROPOSTA DE COMPRA

O objetivo desse apêndice é apresentar o processo “Submeter proposta de compra”, o qual foi o cenário utilizado para a elaboração deste trabalho.

A Tabela 31 apresenta a notação utilizada no macro processo “Comprar imóvel”, no processo “Submeter proposta de compra” e na descrição das atividades do mesmo processo.

**Tabela 31 - Notação dos elementos utilizados na modelagem de processos**

Nome	Semântica	Sintaxe
(Link para outro modelo)	Este símbolo indica que o objeto tem um modelo associado a ele.	
Atividade	Representa um processo ou uma etapa de uma sequência que precisa ser executada para que um processo seja realizado.	
Atividade automatizada (sistêmica)	Representa uma atividade executada automaticamente e exclusivamente por um sistema, sem interferência de uma pessoa. Pessoas, equipamentos, sensores, outros sistemas etc., podem interagir com esta atividade apenas como disparadores ou receptores de resultados.	
Base de dados	Representa informações ou dados que estão armazenadas em banco de dados de sistemas de aplicações.	
Dados / Informação	Representa um conjunto de informações (estruturadas ou não) gerado ou consumido durante a execução do processo.	
Evento final	Representa a circunstância ou status final do processo.	
Evento inicial	Representa uma circunstância ou status que propicia o início do processo.	
Evento intermediário	Representa uma circunstância ou status relevante para o entendimento do processo.	
Interface de processo	Representa a interface entre processos (existentes em algum VAC), indicando que existe uma comunicação entre eles. Em geral, trata-se de uma indicação de outro processo que	

	complementa o fluxo modelado, mas não é objeto principal do modelo em questão.	
Operador lógico Ou (AND/OR)	Operador lógico que representa: - quando dividir o fluxo: que pelo menos um dos caminhos precisa ser percorrido, ou seja, no mínimo um dos eventos destino deve ocorrer (podendo ser mais de um ao mesmo tempo). - quando unir o fluxo: que um ou mais caminhos percorridos inicia a atividade/ processo seguinte, ou seja, um ou mais eventos origem podem ocorrer.	
Posto de trabalho	Representa o posto de trabalho (papel/função) que interage com um processo (produzindo ou consumindo informações).	
Processo inicial	Representa um macroprocesso ou processo inicial de uma sequência da cadeia de valor. Pode representar também um processo superior, a partir do qual existe uma sequência de processos a ele subordinados (ou outra cadeia de valor).	
Regra de Negócio	Diretiva destinada a influenciar ou guiar o comportamento do negócio, como suporte à política de negócio que é formulada em resposta a uma oportunidade.	
Requisito de negócio	Requerimentos provenientes do negócio que irão definir ou restringir aspectos dos sistemas de informação.	
Sistema de aplicação	Representa um sistema de informação que apoia a execução ou executa uma ou mais atividades do processo.	

O processo “Submeter proposta de compra” (Figura 26) corresponde ao processo no qual o cliente pesquisa por um imóvel e cadastra uma proposta de compra. Uma descrição detalhada de suas atividades é apresentada a seguir.

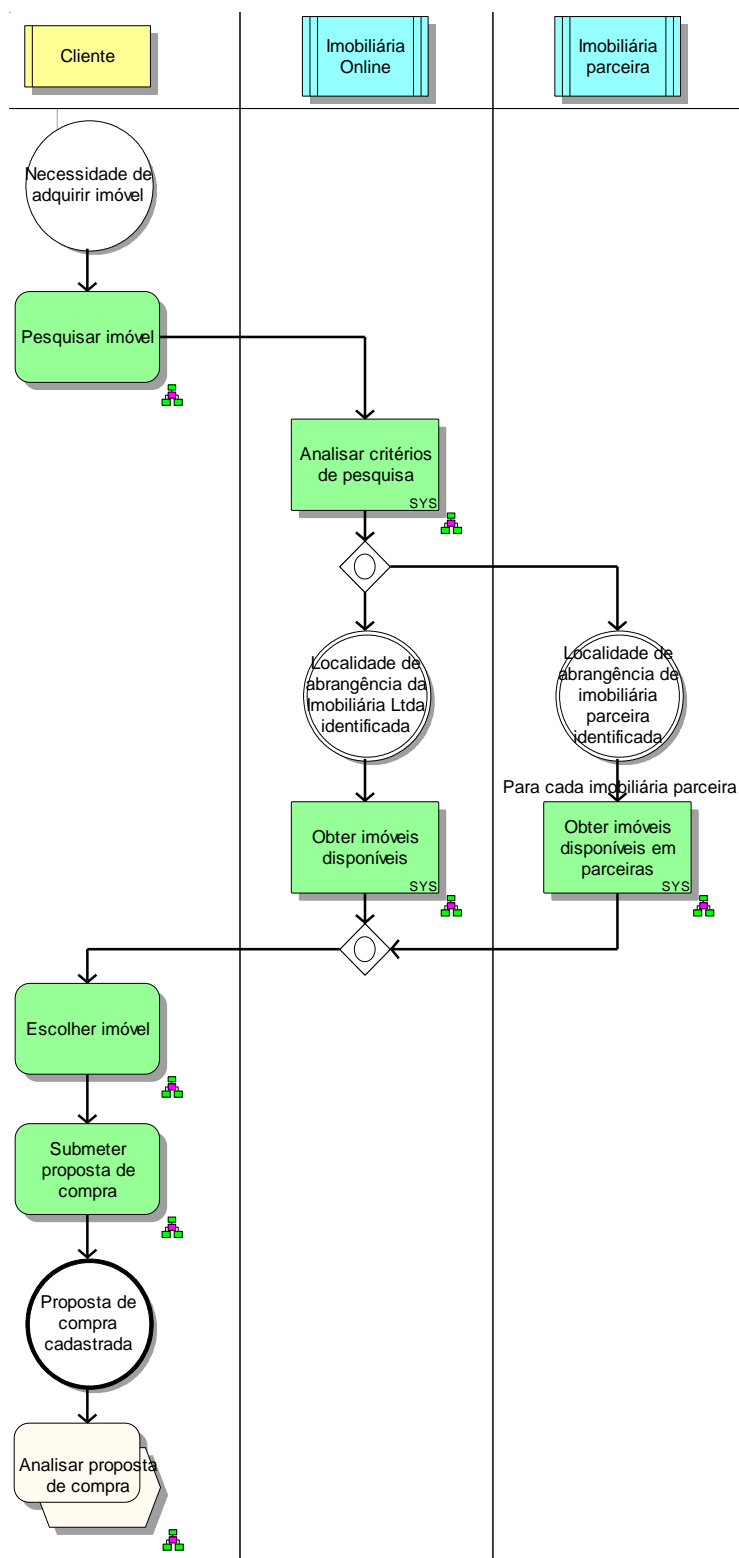


Figura 26 - Processo "Submeter proposta de compra"

## 1. Detalhamento de atividades do processo

### a. Pesquisar imóvel

O cliente registra, utilizando o sistema Imobiliária Online, os critérios que deseja que sejam utilizados para pesquisa por imóveis.

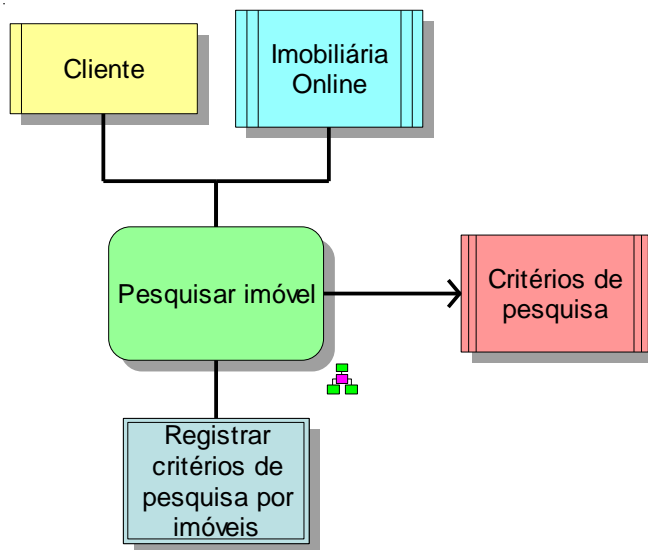


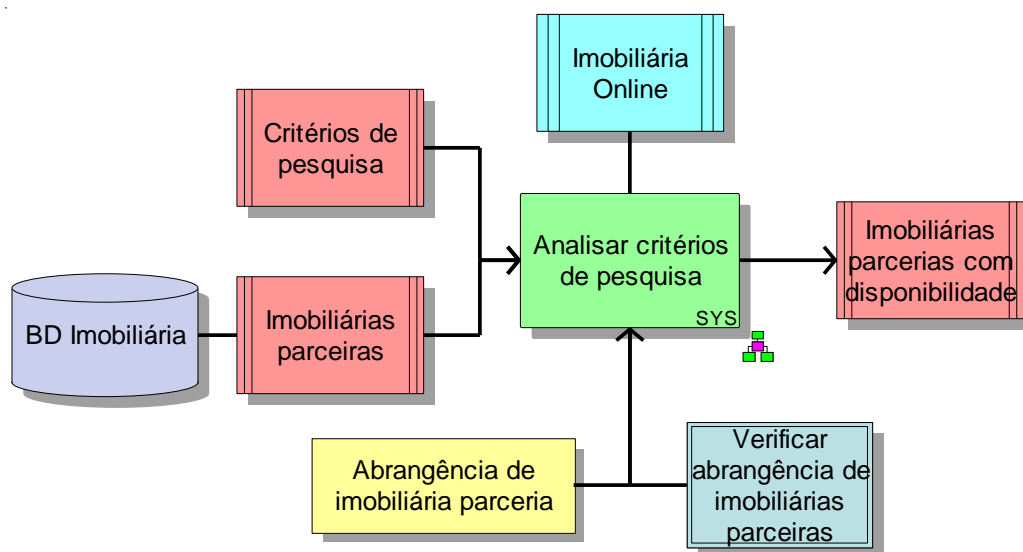
Figura 27 - Pesquisar imóvel

Tabela 32 - Descrição do Requisito de Negócio "Registrar critérios de pesquisa por imóveis"

Requisito de Negócio	Descrição
Registrar critérios de pesquisa por imóveis	O sistema Imobiliária Online deve registrar os critérios do cliente utilizados para pesquisar por imóveis disponíveis: <ul style="list-style-type: none"><li>- CPF do cliente</li><li>- Endereço</li><li>- Faixa de preço</li><li>- Número de quartos</li><li>- Se possui vaga de garagem</li><li>- Se é sol da manhã e/ou da tarde</li></ul>

### b. Analisar critérios de pesquisa

O sistema Imobiliária Online identifica as imobiliárias parceiras que podem ter imóveis disponíveis para serem listados.



**Figura 28 - Analisar critérios de pesquisa**

**Tabela 33 - Descrição do Requisito de Negócio "Verificar abrangência de imobiliárias parceiras"**

Requisito de Negócio	Descrição
Verificar abrangência de imobiliárias parceiras	O sistema Imobiliária Online deve identificar as imobiliárias parceiras que possam atender aos critérios de pesquisa do cliente de acordo com os critérios de pesquisa informados e a regra de negócio "Abrangência de imobiliárias parceiras".

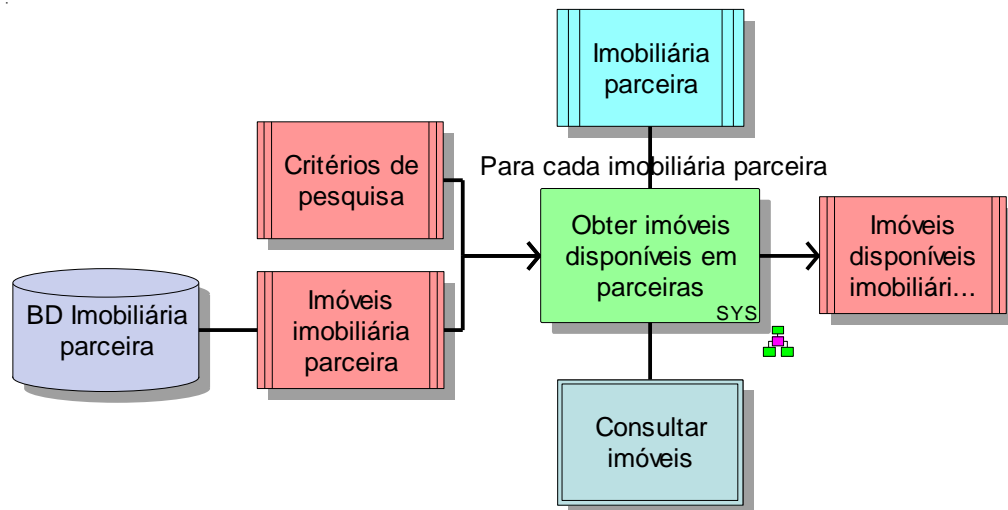
**Tabela 34 - Descrição da Regra de Negócio "Abrangência de imobiliária parceira"**

Regra de Negócio	Descrição
Abrangência de imobiliária parceira	A abrangência de uma imobiliária parceira é definida pelos bairros que ela possui imóvel.

### c. Obter imóveis disponíveis em parceiras

Cada imobiliária parceira retorna seus imóveis disponíveis, de acordo com os critérios de pesquisa.





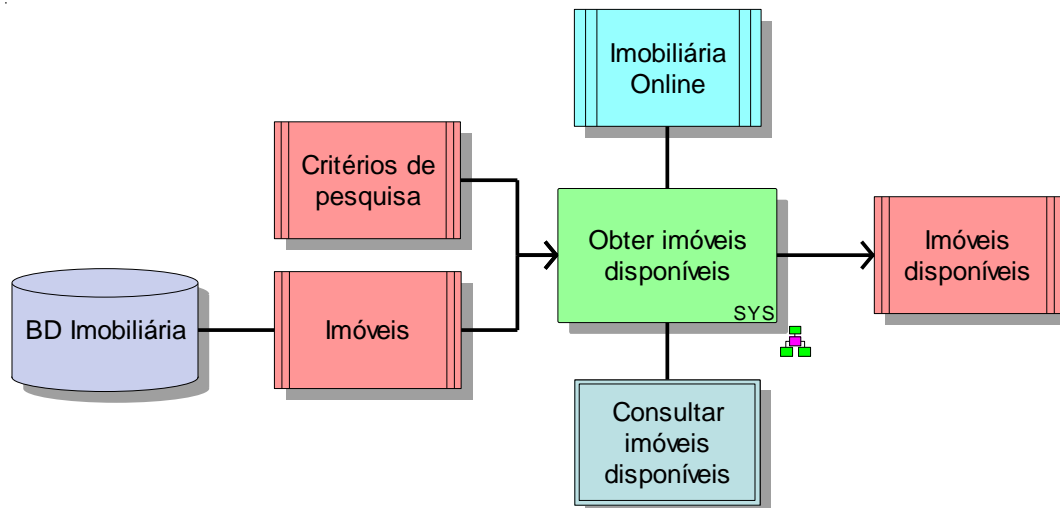
**Figura 29 - Obter imóveis disponíveis em parceiras**

**Tabela 35 - Descrição do Requisito de Negócio "Consultar imóveis"**

Requisito de Negócio	Descrição
Consultar imóveis	<p>O sistema da imobiliária parceira deve consultar os imóveis disponíveis de acordo com os critérios de pesquisa, retornando:</p> <ul style="list-style-type: none"> <li>- Endereço</li> <li>- Preço</li> <li>- Número de quartos</li> <li>- Se possui vaga de garagem</li> <li>- Se é sol da manhã e/ou da tarde</li> <li>- Fotos do imóvel</li> </ul>

#### **d. Obter imóveis disponíveis**

O sistema Imobiliária Online deve consultar os imóveis que possui de acordo com os critérios de pesquisa.



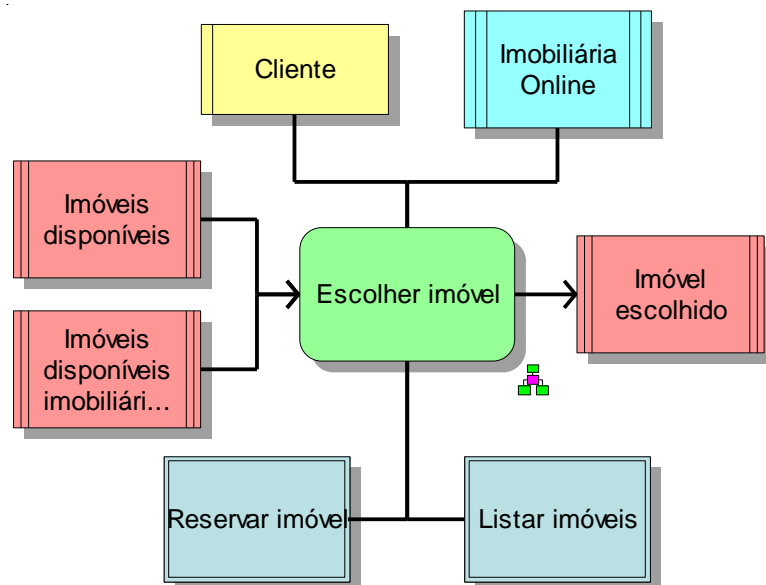
**Figura 30 - Obter imóveis disponíveis**

**Tabela 36 - Descrição do Requisito de Negócio "Consultar imóveis disponíveis"**

Requisito de Negócio	Descrição
Consultar imóveis disponíveis	<p>O sistema Imobiliária Online deve consultar os imóveis disponíveis de acordo com os critérios de pesquisa, retornando:</p> <ul style="list-style-type: none"> <li>- Endereço</li> <li>- Preço</li> <li>- Número de quartos</li> <li>- Se possui vaga de garagem</li> <li>- Se é sol da manhã e/ou da tarde</li> <li>- Fotos do imóvel</li> </ul>

#### **e. Escolher imóvel**

O Cliente escolhe, dentre os imóveis disponíveis na Imobiliária e suas parceiras, o imóvel que atende suas necessidades.



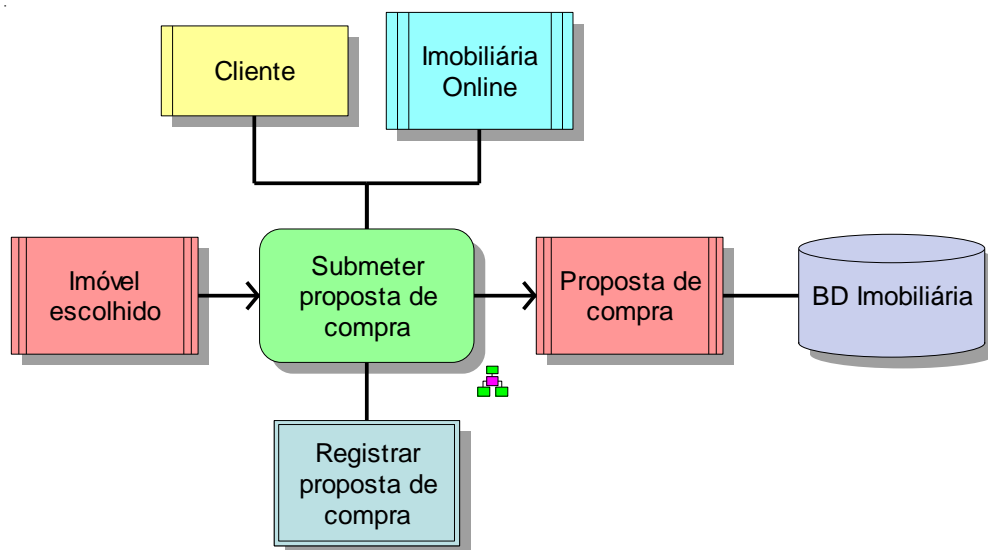
**Figura 31 - Escolher imóvel**

**Tabela 37 - Descrição do Requisito de Negócio "Listar imóveis" e "Reservar imóvel"**

Requisito de Negócio	Descrição
Listar imóveis	<p>O sistema Imobiliária Online deve listar os imóveis disponíveis, exibindo:</p> <ul style="list-style-type: none"> <li>- Endereço</li> <li>- Preço</li> <li>- Número de quartos</li> <li>- Se possui vaga de garagem</li> <li>- Se é sol da manhã e/ou da tarde</li> <li>- Fotos do imóvel</li> </ul>
Reservar imóvel	<p>O sistema Imobiliária Online deve reservar o imóvel escolhido pelo cliente o que pode ocorrer na própria imobiliária ou em imobiliária parceira.</p>

#### **f. Submeter proposta de compra**

A partir do imóvel escolhido, o Cliente cadastra sua proposta de compra.



**Figura 32 - Submeter proposta de compra**

**Tabela 38 - Descrição do Requisito de Negócio "Registrar proposta de compra"**

Requisito de Negócio	Descrição
Registrar proposta de compra	<p>O sistema Imobiliária Online deve registrar a proposta de compra contendo:</p> <ul style="list-style-type: none"> <li>- Nome do cliente</li> <li>- CPF do cliente</li> <li>- Telefone</li> <li>- Endereço</li> <li>- Imóvel escolhido</li> <li>- Imobiliária</li> </ul>

## 2. Informações de apoio ao processo

A Tabela 39 apresenta a descrição das informações consumidas e produzidas no processo “Submeter proposta de compra” que pertence ao macro processo “Comprar imóvel”.

**Tabela 39 - Descrição dos clusters do processo "Submeter proposta de compra"**

Nome do cluster	Descrição do cluster
Crítérios de pesquisa	<p>Representa os critérios de pesquisa para a consulta de imóveis com os seguintes campos:</p> <ul style="list-style-type: none"> <li>- Endereço;</li> <li>- Faixa de preço;</li> <li>- Número de quartos;</li> <li>- Se possui vaga de garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- Data-hora da pesquisa.</li> </ul>

Imobiliárias parceiras	<p>Representa a lista de imobiliárias parceiras e informações de seus imóveis disponíveis a partir do bairro pesquisado, contendo os seguintes campos da imobiliária:</p> <ul style="list-style-type: none"> <li>- Nome da imobiliária;</li> <li>- CNPJ;</li> <li>- Endereço;</li> <li>- Telefones para contato.</li> </ul>
Imobiliárias parceiras com disponibilidade	<p>Representa a lista de imobiliárias parceiras que possuem disponibilidade de imóveis para os critérios de pesquisa selecionados, contendo:</p> <ul style="list-style-type: none"> <li>- Quantidade de imóveis disponíveis que atendem aos critérios de pesquisa;</li> <li>- Informações da imobiliária, como: <ul style="list-style-type: none"> <li>- Nome da imobiliária;</li> <li>- CNPJ;</li> <li>- Endereço;</li> <li>- Telefones para contato.</li> </ul> </li> </ul>
Imóveis imobiliária parceira	<p>Representa a lista de imóveis disponíveis das imobiliárias parceiras, contendo:</p> <ul style="list-style-type: none"> <li>- Informações das imobiliárias parceiras: <ul style="list-style-type: none"> <li>- Nome;</li> <li>- CNPJ;</li> <li>- Endereço;</li> <li>- Telefones para contato;</li> </ul> </li> <li>- Informações dos imóveis, contendo para cada imóvel: <ul style="list-style-type: none"> <li>- Identificador do imóvel;</li> <li>- Endereço;</li> <li>- Preço;</li> <li>- Número de quartos;</li> <li>- Se possui vaga na garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- As fotos do imóvel;</li> <li>- Outros detalhes do imóvel.</li> </ul> </li> </ul>
Imóveis disponíveis imobiliárias parceiras	<p>Representa uma lista dos imóveis disponíveis nas imobiliárias parceiras que atendem aos critérios fornecidos, contendo:</p> <ul style="list-style-type: none"> <li>- Informações das imobiliárias parceiras: <ul style="list-style-type: none"> <li>- Nome;</li> <li>- CNPJ;</li> <li>- Endereço;</li> <li>- Telefones para contato;</li> <li>- Abrangência;</li> </ul> </li> </ul>

	<p>- Informações dos imóveis, contendo para cada imóvel:</p> <ul style="list-style-type: none"> <li>- Identificador do imóvel;</li> <li>- Endereço;</li> <li>- Preço;</li> <li>- Número de quartos;</li> <li>- Se possui vaga na garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- As fotos do imóvel;</li> <li>- Outros detalhes do imóvel.</li> </ul>
Imóveis	<p>Representa os imóveis disponíveis da Imobiliária Online, contendo:</p> <ul style="list-style-type: none"> <li>- Endereço;</li> <li>- Preço;</li> <li>- Número de quartos;</li> <li>- Se possui vaga na garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- As fotos do imóvel;</li> <li>- Outros detalhes do imóvel.</li> </ul>
Imóveis disponíveis	<p>Representa a lista dos imóveis disponíveis na Imobiliária Online que atendem aos critérios fornecidos, contendo para cada imóvel:</p> <ul style="list-style-type: none"> <li>- Identificador do imóvel;</li> <li>- Endereço;</li> <li>- Preço;</li> <li>- Número de quartos;</li> <li>- Se possui vaga na garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- As fotos do imóvel;</li> <li>- Outros detalhes do imóvel.</li> </ul>
Imóvel escolhido	<p>Imóvel selecionado pelo cliente, contendo:</p> <ul style="list-style-type: none"> <li>- Identificador do imóvel;</li> <li>- CNPJ da imobiliária ao qual o imóvel pertence;</li> <li>- Data da escolha do imóvel.</li> </ul>
Proposta de compra	<p>Representa a proposta de compra do cliente contendo:</p> <ul style="list-style-type: none"> <li>- Informações do cliente: <ul style="list-style-type: none"> <li>- Nome do cliente;</li> <li>- CPF do cliente;</li> <li>- Telefone;</li> <li>- Endereço;</li> <li>- E-mail.</li> </ul> </li> <li>- Informações do imóvel escolhido, contendo:</li> </ul>

	<ul style="list-style-type: none"> <li>- CNPJ da imobiliária ao qual o imóvel pertence;</li> <li>- Identificador do imóvel;</li> <li>- Endereço;</li> <li>- Preço;</li> <li>- Número de quartos;</li> <li>- Se possui vaga na garagem;</li> <li>- Se é sol da manhã e/ou sol da tarde;</li> <li>- Valor proposto pelo cliente para a compra do imóvel.</li> <li>- Data da submissão da proposta</li> </ul>
--	--

## APÊNDICE 2 – MODELOS CANÔNICOS

---

Três modelos canônicos foram elaborados, sendo o representado pela Figura 33 o da Imobiliária Online, o da Figura 34 da uma Imobiliária Inglesa e o da Figura 35 da Imobiliária Alemã. O modelo físico referente a estes diagramas de classe é apresentado no Apêndice 3. Estas duas imobiliárias são parceiras da Imobiliária Online e como termo do contrato todas as imobiliárias devem disponibilizar seus dados no padrão apresentado nos modelos canônicos.

Diferente dos modelos das Imobiliárias Inglesa e Alemã no qual um imóvel só pode pertencer a uma imobiliária, no modelo da Imobiliária Online o relacionamento entre imóvel e imobiliária é N para N, representando que um imóvel pode pertencer a uma ou mais imobiliárias. Isso se deve ao fato da imobiliária Online vender imóveis de outras imobiliárias e quando isto ocorre ela deve armazenar os dados referentes ao imóvel da outra imobiliária, resultando assim nesta cardinalidade.

Como um critério de pesquisa pode compreender bairros de imobiliárias parceiras a tabela *Dados*, que se relaciona com a tabela *CritériosPesquisa*, foi elaborada com o objetivo de armazenar os nomes dos bairros pesquisados e o CNPJ da imobiliária ao qual tem abrangência no bairro pesquisado.

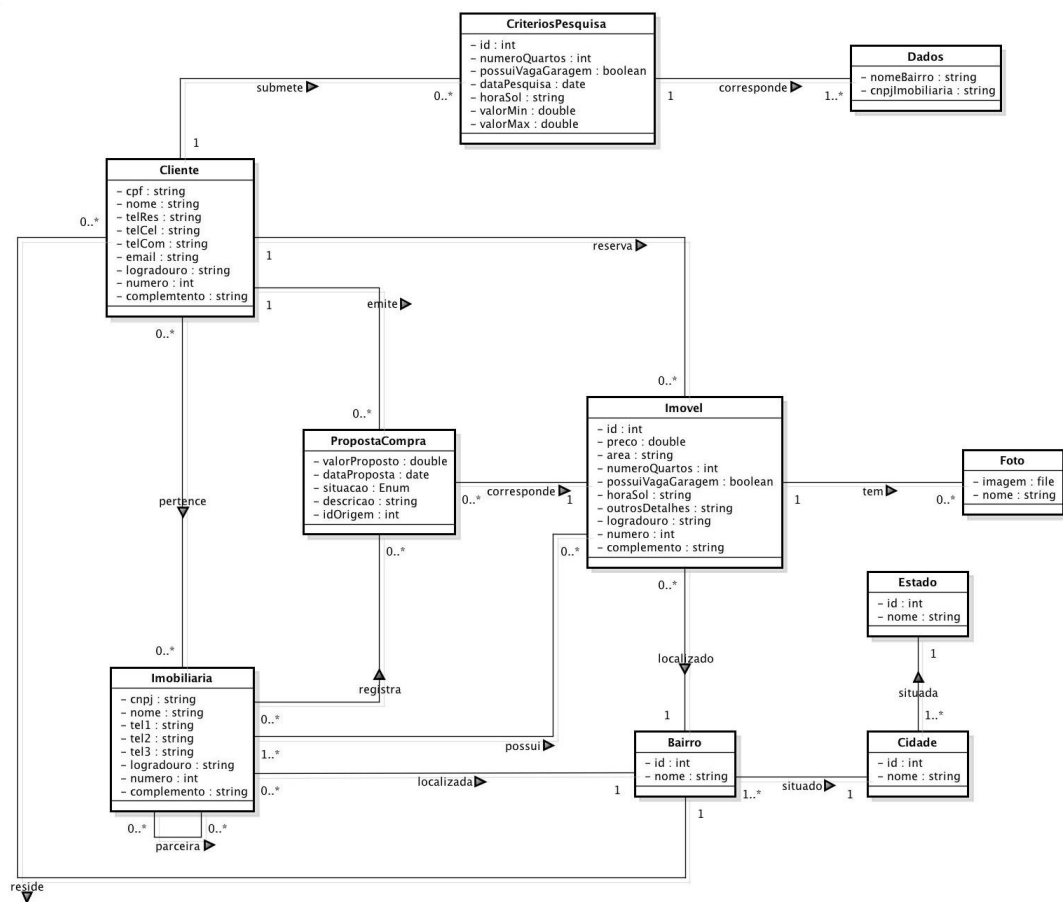
Outra observação é o fato da Imobiliária Inglesa e Alemã possuírem o relacionamento “*scope*” e “*umfasst*” que representam a abrangência de cada imobiliária. Esta abrangência, como descrita na regra de negócio “Abrangência de imobiliária parceira” da atividade “Analisar critérios de pesquisa” apresentada no Apêndice 1 (Figura 28), representa que a abrangência de uma imobiliária parceira é definida pelos bairros que ela possui imóvel.

Observações importantes:

- Faixa de preço é representado por *valorMin* e *valorMax*, onde durante a implementação o usuário pode preencher somente o valor mínimo, ou somente o valor máximo, ou ambos.



- Em proposta de compra há o atributo *situação* do tipo *enum* que armazena a situação da proposta de compra que pode receber os seguintes valores:
  - “*Aguardando\_analise*”: proposta ainda não foi avaliada pela imobiliária.
  - “*Em\_analise*”: proposta sendo avaliada pela imobiliária.
  - “*Negada*”: proposta foi negada pela imobiliária
  - “*Cancelada*”: proposta foi cancelada pelo cliente
  - “*Aprovada*”: proposta foi aprovada pela imobiliária.



**Figura 33 - Modelo canônico da Imobiliária Online**

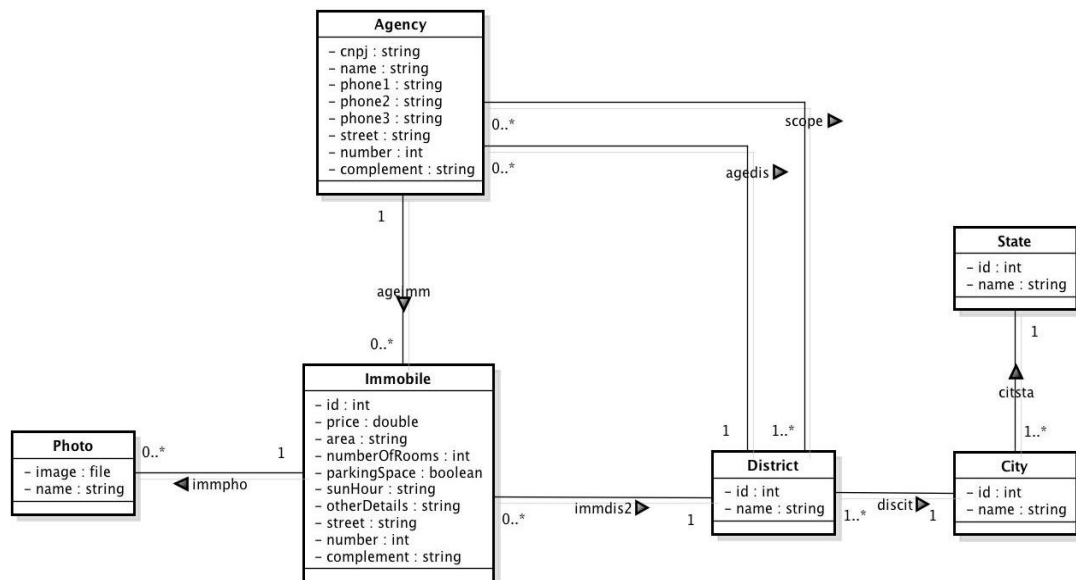


Figura 34 - Modelo canônico da Imobiliária Inglesa

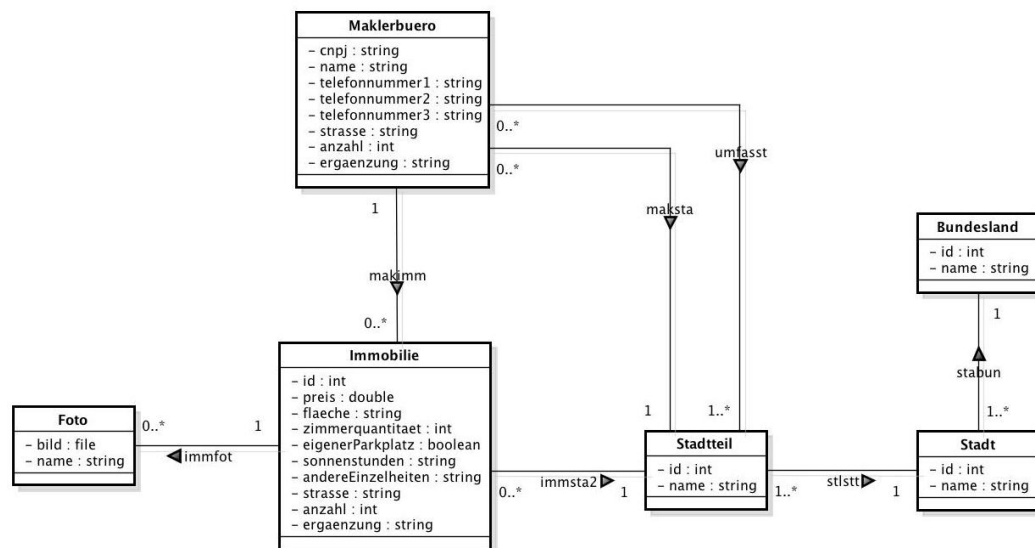


Figura 35 - Modelo canônico da Imobiliária Alemã

## APÊNDICE 3 – MODELOS FÍSICOS

---

Este apêndice tem como objetivo apresentar os scripts para criação dos bancos de dados referentes aos modelos canônicos apresentados no Apêndice 2.

O script para criação do banco de dados da Imobiliária Online, referente ao modelo canônico da Figura 33, é apresentado na Figura 36. O script para criação do banco de dados da Imobiliária Inglesa, referente ao modelo canônico da Figura 34, é apresentado na Figura 37. O script para criação do banco de dados da Imobiliária Alemã, referente ao modelo canônico da Figura 35, é apresentado na Figura 38.

```
--Criação das tabelas
CREATE TABLE IMOBILIARIA
(
    IMB_CNPJ CHARACTER VARYING (50) NOT NULL,
    IMB_NOME CHARACTER VARYING (50) NOT NULL,
    IMB_TEL1 CHARACTER VARYING (50),
    IMB_TEL2 CHARACTER VARYING (50),
    IMB_TEL3 CHARACTER VARYING (50),
    IMB_LOGRADOURO CHARACTER VARYING (100),
    IMB_NUMERO INTEGER,
    IMB_COMPLEMENTO CHARACTER VARYING (100),
    IMB_BRO_ID INTEGER
)
;

ALTER TABLE IMOBILIARIA
    ADD CONSTRAINT IMOBILIARIA_PK PRIMARY KEY ( IMB_CNPJ ) ;

CREATE TABLE PARCEIRA
(
    PAR_IMB_CNPJ1 CHARACTER VARYING (50) NOT NULL,
    PAR_IMB_CNPJ2 CHARACTER VARYING (50) NOT NULL
)
;

ALTER TABLE PARCEIRA
    ADD CONSTRAINT PAR_IMOBILIARIA1_FK FOREIGN KEY
    (
        PAR_IMB_CNPJ1
    )
    REFERENCES IMOBILIARIA
    (
        IMB_CNPJ
    )
;

ALTER TABLE PARCEIRA
    ADD CONSTRAINT PAR_IMOBILIARIA2_FK FOREIGN KEY
    (
```

```

        PAR_IMB_CNPJ2
    )
REFERENCES IMOBILIARIA
(
    IMB_CNPJ
)
;

CREATE TABLE IMOVEL
(
    IMV_ID SERIAL,
    IMV_PRECO DOUBLE PRECISION NOT NULL,
    IMV_AREA CHARACTER VARYING (50) NOT NULL,
    IMV_NUM_QUARTOS INTEGER NOT NULL,
    IMV_POS_VAG_GARAGEM BOOLEAN NOT NULL,
    IMV_HOR_SOL CHARACTER VARYING (20) NOT NULL,
    IMV_OTR_DETALHES CHARACTER VARYING (100),
    IMV_LOGRADOURO CHARACTER VARYING (100) NOT NULL,
    IMV_NUMERO INTEGER NOT NULL,
    IMV_COMPLEMENTO CHARACTER VARYING (100),
    IMV_CLI_CPF CHARACTER VARYING (14),
    IMV_BRO_ID INTEGER
)
;

ALTER TABLE IMOVEL
    ADD CONSTRAINT IMOVEL_PK PRIMARY KEY ( IMV_ID ) ;

CREATE TABLE FOTO
(
    FTO_IMAGEM CHARACTER VARYING (50) NOT NULL,
    FTO_NOME CHARACTER VARYING (100) NOT NULL,
    FTO_IMV_ID INTEGER NOT NULL
)
;

ALTER TABLE FOTO
    ADD CONSTRAINT FTO_IMOVEL_FK FOREIGN KEY
    (
        FTO_IMV_ID
    )
REFERENCES IMOVEL
(
    IMV_ID
)
;

CREATE TABLE CRITERIOS_DE_PESQUISA
(
    CDP_ID SERIAL,
    CDP_NUM_QUARTOS INTEGER NOT NULL,
    CDP_POS_VAG_GARAGEM BOOLEAN NOT NULL,
    CDP_DTA_PESQUISA DATE NOT NULL,
    CDP_HOR_SOL CHARACTER VARYING (20) NOT NULL,
    CDP_VAL_MAX DOUBLE PRECISION NOT NULL,
    CDP_VAL_MIN DOUBLE PRECISION NOT NULL,
    CDP_CLI_CPF CHARACTER VARYING (14) NOT NULL
)
;

```

```

ALTER TABLE CRITERIOS_DE_PESQUISA
    ADD CONSTRAINT CRITERIOS_DE_PESQUISA_PK PRIMARY KEY ( CDP_ID ) ;

CREATE TABLE CLIENTE
(
    CLI_CPF CHARACTER VARYING (14) NOT NULL,
    CLI_NOME CHARACTER VARYING (100) NOT NULL,
    CLI_TEL_RES CHARACTER VARYING (50),
    CLI_TEL_CEL CHARACTER VARYING (50),
    CLI_TEL_COM CHARACTER VARYING (50),
    CLI_EMAIL CHARACTER VARYING (50) NOT NULL,
    CLI_LOGRADOURO CHARACTER VARYING (100) NOT NULL,
    CLI_NUMERO INTEGER NOT NULL,
    CLI_COMPLEMENTO CHARACTER VARYING (100),
    CLI_BRO_ID INTEGER NOT NULL
)
;

ALTER TABLE CLIENTE
    ADD CONSTRAINT CLIENTE_PK PRIMARY KEY ( CLI_CPF ) ;

ALTER TABLE CRITERIOS_DE_PESQUISA
    ADD CONSTRAINT CLI_CLIETNE_FK FOREIGN KEY
    (
        CDP_CLI_CPF
    )
    REFERENCES CLIENTE
    (
        CLI_CPF
    )
;

ALTER TABLE IMOVEL
    ADD CONSTRAINT IMV_CLIENTE_FK FOREIGN KEY
    (
        IMV_CLI_CPF
    )
    REFERENCES CLIENTE
    (
        CLI_CPF
    )
;

CREATE TABLE PROPOSTA_DE_COMPRA
(
    PDC_VALOR DOUBLE PRECISION NOT NULL,
    PDC_DATA DATE NOT NULL,
    PDC_SITUACAO CHARACTER VARYING (50) NOT NULL,
    PDC_DECRICAO CHARACTER VARYING (100),
    PDC_IMB_CNPJ CHARACTER VARYING (50) NOT NULL,
    PDC_ID_ORIGEM INTEGER NOT NULL,
    PDC_CLI_CPF CHARACTER VARYING (14) NOT NULL,
    PDC_IMV_ID INTEGER NOT NULL
)
;

ALTER TABLE PROPOSTA_DE_COMPRA
    ADD CONSTRAINT PDC_IMOBILIARIA_FK FOREIGN KEY

```

```

        (
            PDC_IMB_CNPJ
        )
    REFERENCES IMOBILIARIA
    (
        IMB_CNPJ
    )
;

ALTER TABLE PROPOSTA_DE_COMPRA
    ADD CONSTRAINT PDC_CLIENTE_FK FOREIGN KEY
    (
        PDC_CLI_CPF
    )
    REFERENCES CLIENTE
    (
        CLI_CPF
    )
;

ALTER TABLE PROPOSTA_DE_COMPRA
    ADD CONSTRAINT PDC_IMOVEL_FK FOREIGN KEY
    (
        PDC_IMV_ID
    )
    REFERENCES IMOVEL
    (
        IMV_ID
    )
;

CREATE TABLE PERTENCE
    (
        PRT_CLI_CPF CHARACTER VARYING (14) NOT NULL,
        PRT_IMB_CNPJ CHARACTER VARYING (50) NOT NULL
    )
;

ALTER TABLE PERTENCE
    ADD CONSTRAINT PRT_CLIENTE_FK FOREIGN KEY
    (
        PRT_CLI_CPF
    )
    REFERENCES CLIENTE
    (
        CLI_CPF
    )
;

ALTER TABLE PERTENCE
    ADD CONSTRAINT PRT_IMOBILIARIA_FK FOREIGN KEY
    (
        PRT_IMB_CNPJ
    )
    REFERENCES IMOBILIARIA
    (
        IMB_CNPJ
    )
;

```

```

CREATE TABLE POSSUI
(
    POS_IMV_ID INTEGER NOT NULL,
    POS_IMB_CNPJ CHARACTER VARYING (50) NOT NULL
)
;

CREATE TABLE BAIRRO
(
    BRO_ID SERIAL,
    BRO_NOME CHARACTER VARYING (100) NOT NULL,
    BRO_CID_ID INTEGER NOT NULL
)
;

ALTER TABLE BAIRRO
    ADD CONSTRAINT BAIRRO_PK PRIMARY KEY ( BRO_ID ) ;

CREATE TABLE CIDADE
(
    CID_ID SERIAL,
    CID_NOME CHARACTER VARYING (100) NOT NULL,
    CID_EST_ID INTEGER NOT NULL
)
;

ALTER TABLE CIDADE
    ADD CONSTRAINT CIDADE_PK PRIMARY KEY ( CID_ID ) ;

CREATE TABLE ESTADO
(
    EST_ID SERIAL,
    EST_NOME CHARACTER VARYING (100) NOT NULL
)
;

ALTER TABLE ESTADO
    ADD CONSTRAINT ESTADO_PK PRIMARY KEY ( EST_ID ) ;

CREATE TABLE DADOS
(
    DAD_CDP_ID SERIAL,
    DAD_NOME_BAIRRO CHARACTER VARYING (100) NOT NULL,
    DAD_CNPJ_IMOBILIARIA CHARACTER VARYING (50) NOT NULL
)
;

ALTER TABLE DADOS
    ADD CONSTRAINT DAD_IMOBILIARIA_FK FOREIGN KEY
    (
        DAD_CNPJ_IMOBILIARIA
    )
    REFERENCES IMOBILIARIA
    (
        IMB_CNPJ
    )
;

```

```

ALTER TABLE DADOS
    ADD CONSTRAINT DAD_CRITERIOS_DE_PESQUISA_FK FOREIGN KEY
    (
        DAD_CDP_ID
    )
    REFERENCES CRITERIOS_DE_PESQUISA
    (
        CDP_ID
    )
;

ALTER TABLE IMOVEL
    ADD CONSTRAINT IMV_BAIRRO_FK FOREIGN KEY
    (
        IMV_BRO_ID
    )
    REFERENCES BAIRRO
    (
        BRO_ID
    )
;

ALTER TABLE IMOBILIARIA
    ADD CONSTRAINT IMB_BAIRRO_FK FOREIGN KEY
    (
        IMB_BRO_ID
    )
    REFERENCES BAIRRO
    (
        BRO_ID
    )
;

ALTER TABLE CLIENTE
    ADD CONSTRAINT CLI_BAIRRO_FK FOREIGN KEY
    (
        CLI_BRO_ID
    )
    REFERENCES BAIRRO
    (
        BRO_ID
    )
;

ALTER TABLE POSSUI
    ADD CONSTRAINT POS_IMOVEL_FK FOREIGN KEY
    (
        POS_IMV_ID
    )
    REFERENCES IMOVEL
    (
        IMV_ID
    )
;

ALTER TABLE POSSUI
    ADD CONSTRAINT POS_IMOBILIARIA_FK FOREIGN KEY
    (
        POS_IMB_CNPJ

```



```

    )
    REFERENCES IMOBILIARIA
    (
        IMB_CNPJ
    )
;

--Inserts nas tabelas
INSERT INTO ESTADO (EST_NOME) VALUES ('Rio de Janeiro');

INSERT INTO CIDADE (CID_NOME, CID_EST_ID) VALUES ('Rio de Janeiro',
(select EST_ID from ESTADO where EST_NOME='Rio de Janeiro'));
INSERT INTO CIDADE (CID_NOME, CID_EST_ID) VALUES ('Cabo Frio',
(select EST_ID from ESTADO where EST_NOME='Rio de Janeiro'));

INSERT INTO BAIRRO (BRO_NOME, BRO_CID_ID) VALUES ('Ipanema', (select
CID_ID from CIDADE where CID_NOME='Rio de Janeiro'));
INSERT INTO BAIRRO (BRO_NOME, BRO_CID_ID) VALUES ('Copacabana',
(select CID_ID from CIDADE where CID_NOME='Rio de Janeiro'));
INSERT INTO BAIRRO (BRO_NOME, BRO_CID_ID) VALUES ('Urca', (select
CID_ID from CIDADE where CID_NOME='Rio de Janeiro'));
INSERT INTO BAIRRO (BRO_NOME, BRO_CID_ID) VALUES ('Tijuca', (select
CID_ID from CIDADE where CID_NOME='Rio de Janeiro'));
INSERT INTO BAIRRO (BRO_NOME, BRO_CID_ID) VALUES ('Barra da Tijuca',
(select CID_ID from CIDADE where CID_NOME='Rio de Janeiro'));
INSERT INTO BAIRRO (BRO_NOME, BRO_CID_ID) VALUES ('Centro', (select
CID_ID from CIDADE where CID_NOME='Cabo Frio'));

INSERT INTO IMOBILIARIA (IMB_CNPJ, IMB_NOME, IMB_TEL1,
IMB_LOGRADOURO, IMB_NUMERO, IMB_COMPLEMENTO, IMB_BRO_ID) VALUES
('14.218.935/0001-27', 'Imobiliaria Rio de Janeiro', '(21) 2440-
7766', 'Av Nossa Senhora de Copacabana', 900, 'Loja 2', (select
BRO_ID from BAIRRO where BRO_NOME='Copacabana'));
INSERT INTO IMOBILIARIA (IMB_CNPJ, IMB_NOME, IMB_TEL1) VALUES
('14.218.835/0001-27', 'Imobiliaria Londres', '(11) 3033-4878');
INSERT INTO IMOBILIARIA (IMB_CNPJ, IMB_NOME, IMB_TEL1) VALUES
('03.847.655/0001-98', 'Imobiliaria Berlim', '(85) 2273-0135');

INSERT INTO PARCEIRA (PAR_IMB_CNPJ1, PAR_IMB_CNPJ2) VALUES
('14.218.935/0001-27', '14.218.835/0001-27');
INSERT INTO PARCEIRA (PAR_IMB_CNPJ1, PAR_IMB_CNPJ2) VALUES
('14.218.935/0001-27', '03.847.655/0001-98');

INSERT INTO IMOVEL (IMV_PRECO, IMV_AREA, IMV_NUM_QUARTOS,
IMV_POS_VAG_GARAGEM, IMV_HOR_SOL, IMV_LOGRADOURO, IMV_NUMERO,
IMV_COMPLEMENTO, IMV_BRO_ID) VALUES (1800000.00, '150m2', 2, true,
'Manha', 'Av Vieira Souto', 396, 'AP101', (select BRO_ID from BAIRRO
inner join CIDADE on BRO_CID_ID=CID_ID where BRO_NOME='Ipanema' and
CID_NOME='Rio de Janeiro'));
INSERT INTO IMOVEL (IMV_PRECO, IMV_AREA, IMV_NUM_QUARTOS,
IMV_POS_VAG_GARAGEM, IMV_HOR_SOL, IMV_LOGRADOURO, IMV_NUMERO,
IMV_COMPLEMENTO, IMV_BRO_ID) VALUES (1500000.00, '180m2', 2, true,
'Manha', 'Av Atlantica', 800, 'AP104', (select BRO_ID from BAIRRO
inner join CIDADE on BRO_CID_ID=CID_ID where BRO_NOME='Copacabana'
and CID_NOME='Rio de Janeiro'));
INSERT INTO IMOVEL (IMV_PRECO, IMV_AREA, IMV_NUM_QUARTOS,
IMV_POS_VAG_GARAGEM, IMV_HOR_SOL, IMV_LOGRADOURO, IMV_NUMERO,
IMV_COMPLEMENTO, IMV_BRO_ID) VALUES (650000.00, '110m2', 2, true,

```

```

'Manha', 'Rua Haddock Lobo', 400, 'AP707', (select BRO_ID from
BAIRRO inner join CIDADE on BRO_CID_ID=CID_ID where
BRO_NOME='Tijuca' and CID_NOME='Rio de Janeiro'));
INSERT INTO IMOVEL (IMV_PRECO, IMV_AREA, IMV_NUM_QUARTOS,
IMV_POS_VAG_GARAGEM, IMV_HOR_SOL, IMV_LOGRADOURO, IMV_NUMERO,
IMV_COMPLEMENTO, IMV_BRO_ID) VALUES (810000.00, '115m2', 3, true,
'Tarde', 'Avenida Erico Verissimo', 656, 'AP303', (select BRO_ID
from BAIRRO inner join CIDADE on BRO_CID_ID=CID_ID where
BRO_NOME='Barra da Tijuca' and CID_NOME='Rio de Janeiro'));
INSERT INTO IMOVEL (IMV_PRECO, IMV_AREA, IMV_NUM_QUARTOS,
IMV_POS_VAG_GARAGEM, IMV_HOR_SOL, IMV_LOGRADOURO, IMV_NUMERO,
IMV_COMPLEMENTO, IMV_BRO_ID) VALUES (480000.00, '65m2', 1, false,
'Manha', 'Avenida Francisco Mendes', 285, 'Casa', (select BRO_ID
from BAIRRO inner join CIDADE on BRO_CID_ID=CID_ID where
BRO_NOME='Centro' and CID_NOME='Cabo Frio'));

INSERT INTO POSSUI (POS_IMV_ID, POS_IMB_CNPJ) VALUES ((select IMV_ID
from IMOVEL where IMV_PRECO=1800000.00 and IMV_AREA='150m2' and
IMV_BRO_ID=(select BRO_ID from BAIRRO inner join CIDADE on
BRO_CID_ID=CID_ID where BRO_NOME='Ipanema' and CID_NOME='Rio de
Janeiro')), (select IMB_CNPJ from IMOBILIARIA where
IMB_CNPJ='14.218.935/0001-27'));
INSERT INTO POSSUI (POS_IMV_ID, POS_IMB_CNPJ) VALUES ((select IMV_ID
from IMOVEL where IMV_PRECO=1500000.00 and IMV_AREA='180m2' and
IMV_BRO_ID=(select BRO_ID from BAIRRO inner join CIDADE on
BRO_CID_ID=CID_ID where BRO_NOME='Copacabana' and CID_NOME='Rio de
Janeiro')), (select IMB_CNPJ from IMOBILIARIA where
IMB_CNPJ='14.218.935/0001-27'));
INSERT INTO POSSUI (POS_IMV_ID, POS_IMB_CNPJ) VALUES ((select IMV_ID
from IMOVEL where IMV_PRECO=650000.00 and IMV_AREA='110m2' and
IMV_BRO_ID=(select BRO_ID from BAIRRO inner join CIDADE on
BRO_CID_ID=CID_ID where BRO_NOME='Tijuca' and CID_NOME='Rio de
Janeiro')), (select IMB_CNPJ from IMOBILIARIA where
IMB_CNPJ='14.218.935/0001-27'));
INSERT INTO POSSUI (POS_IMV_ID, POS_IMB_CNPJ) VALUES ((select IMV_ID
from IMOVEL where IMV_PRECO=810000.00 and IMV_AREA='115m2' and
IMV_BRO_ID=(select BRO_ID from BAIRRO inner join CIDADE on
BRO_CID_ID=CID_ID where BRO_NOME='Barra da Tijuca' and CID_NOME='Rio
de Janeiro')), (select IMB_CNPJ from IMOBILIARIA where
IMB_CNPJ='14.218.935/0001-27'));
INSERT INTO POSSUI (POS_IMV_ID, POS_IMB_CNPJ) VALUES ((select IMV_ID
from IMOVEL where IMV_PRECO=480000.00 and IMV_AREA='65m2' and
IMV_BRO_ID=(select BRO_ID from BAIRRO inner join CIDADE on
BRO_CID_ID=CID_ID where BRO_NOME='Centro' and CID_NOME='Cabo
Frio')), (select IMB_CNPJ from IMOBILIARIA where
IMB_CNPJ='14.218.935/0001-27'));

INSERT INTO CLIENTE (CLI_CPF, CLI_NOME, CLI_TEL_RES, CLI_EMAIL,
CLI_LOGRADOURO, CLI_NUMERO, CLI_COMPLEMENTO, CLI_BRO_ID) VALUES
('111.222.333-45', 'Joao', '4040-7077', 'joao@joao.com.br', 'Av
Pasteur', 459, 'Bl1 AP15000', (select BRO_ID from BAIRRO where
BRO_NOME='Urca'));

```

**Figura 36 - Modelo físico da Imobiliária Online**

```

--Criação das tabelas
CREATE TABLE PHOTO
(
    PHO_IMAGE CHARACTER VARYING (50) NOT NULL,

```

```

        PHO_NAME CHARACTER VARYING (100) NOT NULL,
        PHO_IMM_ID INTEGER NOT NULL
    )
;

CREATE TABLE AGENCY
(
    AGE_CNPJ CHARACTER VARYING (50) NOT NULL,
    AGE_NAME CHARACTER VARYING (100) NOT NULL,
    AGE_PHONE1 CHARACTER VARYING (50),
    AGE_PHONE2 CHARACTER VARYING (50),
    AGE_PHONE3 CHARACTER VARYING (50),
    AGE_STREET CHARACTER VARYING (100),
    AGE_NUMBER INTEGER,
    AGE_COMPLEMENT CHARACTER VARYING (100),
    AGE_DIS_ID INTEGER
)
;

ALTER TABLE AGENCY
    ADD CONSTRAINT AGENCY_PK PRIMARY KEY ( AGE_CNPJ ) ;

CREATE TABLE IMMOBILE
(
    IMM_ID SERIAL,
    IMM_PRICE DOUBLE PRECISION NOT NULL,
    IMM_AREA CHARACTER VARYING (50) NOT NULL,
    IMM_NUM_ROOMS INTEGER NOT NULL,
    IMM_PRK_SPACE BOOLEAN NOT NULL,
    IMM_SUN_HOUR CHARACTER VARYING (20) NOT NULL,
    IMM_OTR_DETAILS CHARACTER VARYING (100),
    IMM_STREET CHARACTER VARYING (100) NOT NULL,
    IMM_NUMBER INTEGER NOT NULL,
    IMM_COMPLEMENT CHARACTER VARYING (100),
    IMM_AGE_CNPJ CHARACTER VARYING (50) NOT NULL,
    IMM_DIS_ID INTEGER NOT NULL
)
;

ALTER TABLE IMMOBILE
    ADD CONSTRAINT IMMOBILE_PK PRIMARY KEY ( IMM_ID ) ;

ALTER TABLE IMMOBILE
    ADD CONSTRAINT IMM_AGENCY_FK FOREIGN KEY
    (
        IMM_AGE_CNPJ
    )
    REFERENCES AGENCY
    (
        AGE_CNPJ
    )
;

CREATE TABLE DISTRICT
(
    DIS_ID SERIAL,
    DIS_NAME CHARACTER VARYING (100) NOT NULL,
    DIS_CIT_ID INTEGER NOT NULL
)

```

```

;

ALTER TABLE DISTRICT
    ADD CONSTRAINT DISTRICT_PK PRIMARY KEY ( DIS_ID ) ;

CREATE TABLE CITY
(
    CIT_ID SERIAL,
    CIT_NAME CHARACTER VARYING (100) NOT NULL,
    CIT_STA_ID INTEGER NOT NULL
)
;

ALTER TABLE CITY
    ADD CONSTRAINT CITY_PK PRIMARY KEY ( CIT_ID ) ;

CREATE TABLE STATE
(
    STA_ID SERIAL,
    STA_NAME CHARACTER VARYING (100) NOT NULL
)
;

ALTER TABLE STATE
    ADD CONSTRAINT STATE_PK PRIMARY KEY ( STA_ID ) ;

CREATE TABLE SCOPE
(
    SCO_DIS_ID INTEGER NOT NULL,
    SCO_AGE_CNPJ CHARACTER VARYING (50) NOT NULL
)
;

ALTER TABLE SCOPE
    ADD CONSTRAINT SCO_DISTRICT_FK FOREIGN KEY
    (
        SCO_DIS_ID
    )
    REFERENCES DISTRICT
    (
        DIS_ID
    )
;

ALTER TABLE SCOPE
    ADD CONSTRAINT SCO_AGENCY_FK FOREIGN KEY
    (
        SCO_AGE_CNPJ
    )
    REFERENCES AGENCY
    (
        AGE_CNPJ
    )
;

ALTER TABLE PHOTO
    ADD CONSTRAINT PHO_IMMOBILE_FK FOREIGN KEY
    (
        PHO_IMM_ID

```

```

    )
    REFERENCES IMMOBILE
    (
        IMM_ID
    )
;

ALTER TABLE AGENCY
    ADD CONSTRAINT AGE_DISTRICT_FK FOREIGN KEY
    (
        AGE_DIS_ID
    )
    REFERENCES DISTRICT
    (
        DIS_ID
    )
;

ALTER TABLE IMMOBILE
    ADD CONSTRAINT IMM_DISTRICT_FK FOREIGN KEY
    (
        IMM_DIS_ID
    )
    REFERENCES DISTRICT
    (
        DIS_ID
    )
;

ALTER TABLE DISTRICT
    ADD CONSTRAINT DIS_CITY_FK FOREIGN KEY
    (
        DIS_CIT_ID
    )
    REFERENCES CITY
    (
        CIT_ID
    )
;

ALTER TABLE CITY
    ADD CONSTRAINT CIT_STATE_FK FOREIGN KEY
    (
        CIT_STA_ID
    )
    REFERENCES STATE
    (
        STA_ID
    )
;

--Inserts nas tabelas
INSERT INTO STATE (STA_NAME) VALUES ('Sao Paulo');

INSERT INTO CITY (CIT_NAME, CIT_STA_ID) VALUES ('Sao Paulo', (select
STA_ID from STATE where STA_NAME='Sao Paulo'));
INSERT INTO CITY (CIT_NAME, CIT_STA_ID) VALUES ('Guarulhos', (select
STA_ID from STATE where STA_NAME='Sao Paulo'));

```

```

INSERT INTO CITY (CIT_NAME, CIT_STA_ID) VALUES ('Praia Grande',
(select STA_ID from STATE where STA_NAME='Sao Paulo'));

INSERT INTO DISTRICT (DIS_NAME, DIS_CIT_ID) VALUES ('Jardim
Paulista', (select CIT_ID from CITY where CIT_NAME='Sao Paulo'));
INSERT INTO DISTRICT (DIS_NAME, DIS_CIT_ID) VALUES ('Campo Belo',
(select CIT_ID from CITY where CIT_NAME='Sao Paulo'));
INSERT INTO DISTRICT (DIS_NAME, DIS_CIT_ID) VALUES ('Bom Retiro',
(select CIT_ID from CITY where CIT_NAME='Sao Paulo'));
INSERT INTO DISTRICT (DIS_NAME, DIS_CIT_ID) VALUES ('Jardim
Paraventi', (select CIT_ID from CITY where CIT_NAME='Guarulhos'));
INSERT INTO DISTRICT (DIS_NAME, DIS_CIT_ID) VALUES ('Vila Tupi',
(select CIT_ID from CITY where CIT_NAME='Praia Grande'));

INSERT INTO AGENCY (AGE_CNPJ, AGE_NAME, AGE_PHONE1, AGE_STREET,
AGE_NUMBER, AGE_COMPLEMENT, AGE_DIS_ID) VALUES ('14.218.835/0001-
27', 'Imobiliária Londres', '(11) 3033-4878', 'Av Angelica', 475,
'Loja 7', (select DIS_ID from DISTRICT inner join CITY on
DIS_CIT_ID=CIT_ID where DIS_NAME='Bom Retiro' and CIT_NAME='Sao
Paulo'));

INSERT INTO SCOPE (SCO_DIS_ID, SCO_AGE_CNPJ) VALUES ((select DIS_ID
from DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where
DIS_NAME='Jardim Paulista' and CIT_NAME='Sao Paulo'), (select
AGE_CNPJ from AGENCY where AGE_CNPJ='14.218.835/0001-27'));
INSERT INTO SCOPE (SCO_DIS_ID, SCO_AGE_CNPJ) VALUES ((select DIS_ID
from DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where
DIS_NAME='Campo Belo' and CIT_NAME='Sao Paulo'), (select AGE_CNPJ
from AGENCY where AGE_CNPJ='14.218.835/0001-27'));
INSERT INTO SCOPE (SCO_DIS_ID, SCO_AGE_CNPJ) VALUES ((select DIS_ID
from DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where
DIS_NAME='Bom Retiro' and CIT_NAME='Sao Paulo'), (select AGE_CNPJ
from AGENCY where AGE_CNPJ='14.218.835/0001-27'));
INSERT INTO SCOPE (SCO_DIS_ID, SCO_AGE_CNPJ) VALUES ((select DIS_ID
from DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where
DIS_NAME='Jardim Paraventi' and CIT_NAME='Guarulhos'), (select
AGE_CNPJ from AGENCY where AGE_CNPJ='14.218.835/0001-27'));
INSERT INTO SCOPE (SCO_DIS_ID, SCO_AGE_CNPJ) VALUES ((select DIS_ID
from DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where
DIS_NAME='Vila Tupi' and CIT_NAME='Praia Grande'), (select AGE_CNPJ
from AGENCY where AGE_CNPJ='14.218.835/0001-27'));

INSERT INTO IMMOBILE (IMM_PRICE, IMM_AREA, IMM_NUM_ROOMS,
IMM_PRK_SPACE, IMM_SUN_HOUR, IMM_STREET, IMM_NUMBER, IMM_COMPLEMENT,
IMM_AGE_CNPJ, IMM_DIS_ID) VALUES (960000.00, '150m2', 3, true,
'Manha', 'Alameda Itu', 546, 'BL2 AP1002', (select AGE_CNPJ from
AGENCY where AGE_CNPJ='14.218.835/0001-27'), (select DIS_ID from
DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where DIS_NAME='Jardim
Paulista' and CIT_NAME='Sao Paulo'));
INSERT INTO IMMOBILE (IMM_PRICE, IMM_AREA, IMM_NUM_ROOMS,
IMM_PRK_SPACE, IMM_SUN_HOUR, IMM_STREET, IMM_NUMBER, IMM_COMPLEMENT,
IMM_AGE_CNPJ, IMM_DIS_ID) VALUES (425000.00, '70m2', 1, false,
'Tarde', 'Rua Doutor Jesuino Maciel', 1063, 'BL8 AP115', (select
AGE_CNPJ from AGENCY where AGE_CNPJ='14.218.835/0001-27'), (select
DIS_ID from DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where
DIS_NAME='Campo Belo' and CIT_NAME='Sao Paulo'));
INSERT INTO IMMOBILE (IMM_PRICE, IMM_AREA, IMM_NUM_ROOMS,
IMM_PRK_SPACE, IMM_SUN_HOUR, IMM_STREET, IMM_NUMBER, IMM_COMPLEMENT,
IMM_AGE_CNPJ, IMM_DIS_ID) VALUES (990000.00, '350m2', 4, true,

```

```

'Manha', 'Rua Apa', 366, 'Casa', (select AGE_CNPJ from AGENCY where
AGE_CNPJ='14.218.835/0001-27'), (select DIS_ID from DISTRICT inner
join CITY on DIS_CIT_ID=CIT_ID where DIS_NAME='Bom Retiro' and
CIT_NAME='Sao Paulo')));
INSERT INTO IMMOBILE (IMM_PRICE, IMM_AREA, IMM_NUM_ROOMS,
IMM_PRK_SPACE, IMM_SUN_HOUR, IMM_STREET, IMM_NUMBER, IMM_COMPLEMENT,
IMM_AGE_CNPJ, IMM_DIS_ID) VALUES (1200000.00, '250m2', 2, true,
'Manha', 'Rua Sousa', 366, 'BL1 AP1402', (select AGE_CNPJ from
AGENCY where AGE_CNPJ='14.218.835/0001-27'), (select DIS_ID from
DISTRICT inner join CITY on DIS_CIT_ID=CIT_ID where DIS_NAME='Jardim
Paraventi' and CIT_NAME='Guarulhos')));
INSERT INTO IMMOBILE (IMM_PRICE, IMM_AREA, IMM_NUM_ROOMS,
IMM_PRK_SPACE, IMM_SUN_HOUR, IMM_STREET, IMM_NUMBER, IMM_COMPLEMENT,
IMM_AGE_CNPJ, IMM_DIS_ID) VALUES (730000.00, '106m2', 2, true,
'Tarde', 'Rua Tamoios', 3796, 'Casa', (select AGE_CNPJ from AGENCY
where AGE_CNPJ='14.218.835/0001-27'), (select DIS_ID from DISTRICT
inner join CITY on DIS_CIT_ID=CIT_ID where DIS_NAME='Vila Tupi' and
CIT_NAME='Praia Grande')));

```

**Figura 37 - Modelo físico da Imobiliária Inglesa**

```

--Criação das tabelas
CREATE TABLE FOTO
(
    FTO_BILD CHARACTER VARYING (50) NOT NULL,
    FTO_NAME CHARACTER VARYING (100) NOT NULL,
    FTO_IMM_ID INTEGER NOT NULL
)
;

CREATE TABLE MAKLERBUERO
(
    MAK_CNPJ CHARACTER VARYING (50) NOT NULL,
    MAK_NAME CHARACTER VARYING (100) NOT NULL,
    MAK_TELEFONNUMMER1 CHARACTER VARYING (50),
    MAK_TELEFONNUMMER2 CHARACTER VARYING (50),
    MAK_TELEFONNUMMER3 CHARACTER VARYING (50),
    MAK_STRASSE CHARACTER VARYING (100),
    MAK_ANZAHL INTEGER,
    MAK_ERGARNZUNG CHARACTER VARYING (100),
    MAK_STL_ID INTEGER
)
;

ALTER TABLE MAKLERBUERO
    ADD CONSTRAINT MAKLERBUERO_PK PRIMARY KEY ( MAK_CNPJ ) ;

CREATE TABLE IMMOBILIE
(
    IMM_ID SERIAL,
    IMM_PREIS DOUBLE PRECISION NOT NULL,
    IMM_FLAECHEN CHARACTER VARYING (50) NOT NULL,
    IMM_ZIMMERQUANTITAET INTEGER NOT NULL,
    IMM_EGR_PARKPLATZ BOOLEAN NOT NULL,
    IMM_SONNENSTUNDEN CHARACTER VARYING (20) NOT NULL,
    IMM_ADE_EINZELHEITEN CHARACTER VARYING (100),
    IMM_STRASSE CHARACTER VARYING (100) NOT NULL,
    IMM_ANZAHL INTEGER NOT NULL,
    IMM_ERGARNZUNG CHARACTER VARYING (100),

```

```

        IMM_MAK_CNPJ CHARACTER VARYING (50) NOT NULL,
        IMM_STL_ID INTEGER NOT NULL
    )
;

ALTER TABLE IMMOBILIE
    ADD CONSTRAINT IMMOBILIE_PK PRIMARY KEY ( IMM_ID ) ;

ALTER TABLE IMMOBILIE
    ADD CONSTRAINT IMM_MAKLERBUERO_FK FOREIGN KEY
    (
        IMM_MAK_CNPJ
    )
    REFERENCES MAKLERBUERO
    (
        MAK_CNPJ
    )
;

CREATE TABLE STADTTEIL
    (
        STL_ID SERIAL,
        STL_NAME CHARACTER VARYING (100) NOT NULL,
        STL_STT_ID INTEGER NOT NULL
    )
;

ALTER TABLE STADTTEIL
    ADD CONSTRAINT STADTTEIL_PK PRIMARY KEY ( STL_ID ) ;

CREATE TABLE STADT
    (
        STT_ID SERIAL,
        STT_NAME CHARACTER VARYING (100) NOT NULL,
        STT_BUN_ID INTEGER NOT NULL
    )
;

ALTER TABLE STADT
    ADD CONSTRAINT STADT_PK PRIMARY KEY ( STT_ID ) ;

CREATE TABLE BUNDESLAND
    (
        BUN_ID SERIAL,
        BUN_NAME CHARACTER VARYING (100) NOT NULL
    )
;

ALTER TABLE BUNDESLAND
    ADD CONSTRAINT BUNDESLAND_PK PRIMARY KEY ( BUN_ID ) ;

CREATE TABLE UMFASST
    (
        UMF_STL_ID INTEGER NOT NULL,
        UMF_MAK_CNPJ CHARACTER VARYING (50) NOT NULL
    )
;

ALTER TABLE UMFASST

```



```

        ADD CONSTRAINT UMF_STADTTEIL_FK FOREIGN KEY
        (
            UMF_STL_ID
        )
        REFERENCES STADTTEIL
        (
            STL_ID
        )
    ;

ALTER TABLE UMFASST
    ADD CONSTRAINT UMF_IMMOBILIE_FK FOREIGN KEY
    (
        UMF_MAK_CNPJ
    )
    REFERENCES MAKLERBUERO
    (
        MAK_CNPJ
    )
    ;

ALTER TABLE FOTO
    ADD CONSTRAINT FTO_IMMOBILIE_FK FOREIGN KEY
    (
        FTO_IMM_ID
    )
    REFERENCES IMMOBILIE
    (
        IMM_ID
    )
    ;

ALTER TABLE MAKLERBUERO
    ADD CONSTRAINT MAK_STADTTEIL_FK FOREIGN KEY
    (
        MAK_STL_ID
    )
    REFERENCES STADTTEIL
    (
        STL_ID
    )
    ;

ALTER TABLE IMMOBILIE
    ADD CONSTRAINT IMM_STADTTEIL_FK FOREIGN KEY
    (
        IMM_STL_ID
    )
    REFERENCES STADTTEIL
    (
        STL_ID
    )
    ;

ALTER TABLE STADTTEIL
    ADD CONSTRAINT STL_STADT_FK FOREIGN KEY
    (
        STL_STT_ID
    )

```

```

REFERENCES STADT
(
    STT_ID
)
;

ALTER TABLE STADT
ADD CONSTRAINT STT_BUNDESLAND_FK FOREIGN KEY
(
    STT_BUN_ID
)
REFERENCES BUNDESLAND
(
    BUN_ID
)
;

--Inserts nas tabelas
INSERT INTO BUNDESLAND (BUN_NAME) VALUES ('Ceara');

INSERT INTO STADT (STT_NAME, STT_BUN_ID) VALUES ('Fortaleza',
(select BUN_ID from BUNDESLAND where BUN_NAME='Ceara'));
INSERT INTO STADT (STT_NAME, STT_BUN_ID) VALUES ('Aquiraz', (select
BUN_ID from BUNDESLAND where BUN_NAME='Ceara'));
INSERT INTO STADT (STT_NAME, STT_BUN_ID) VALUES ('Eusebio', (select
BUN_ID from BUNDESLAND where BUN_NAME='Ceara'));

INSERT INTO STADTTEIL (STL_NAME, STL_STT_ID) VALUES ('Aldeota',
(select STT_ID from STADT where STT_NAME='Fortaleza'));
INSERT INTO STADTTEIL (STL_NAME, STL_STT_ID) VALUES ('Meireles',
(select STT_ID from STADT where STT_NAME='Fortaleza'));
INSERT INTO STADTTEIL (STL_NAME, STL_STT_ID) VALUES ('Praia de
Iracema', (select STT_ID from STADT where STT_NAME='Fortaleza'));
INSERT INTO STADTTEIL (STL_NAME, STL_STT_ID) VALUES ('Porto das
Dunas', (select STT_ID from STADT where STT_NAME='Aquiraz'));
INSERT INTO STADTTEIL (STL_NAME, STL_STT_ID) VALUES ('Coite',
(select STT_ID from STADT where STT_NAME='Eusebio'));

INSERT INTO MAKLERBUERO (MAK_CNPJ, MAK_NAME, MAK_TELEFONNUMMER1,
MAK_STRASSE, MAK_ANZAHL, MAK_ERGARNZUNG, MAK_STL_ID) VALUES
('03.847.655/0001-98', 'Imobiliaria Berlim', '(85) 2273-0135', 'Av
Beira Mar', 445, 'Loja 6', (select STL_ID from STADTTEIL inner join
STADT on STL_STT_ID=STT_ID where STL_NAME='Praia de Iracema' and
STT_NAME='Fortaleza'));

INSERT INTO UMFASST (UMF_STL_ID, UMF_MAK_CNPJ) VALUES ((select
STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID where
STL_NAME='Aldeota' and STT_NAME='Fortaleza'), (select MAK_CNPJ from
MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'));
INSERT INTO UMFASST (UMF_STL_ID, UMF_MAK_CNPJ) VALUES ((select
STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID where
STL_NAME='Meireles' and STT_NAME='Fortaleza'), (select MAK_CNPJ from
MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'));
INSERT INTO UMFASST (UMF_STL_ID, UMF_MAK_CNPJ) VALUES ((select
STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID where
STL_NAME='Praia de Iracema' and STT_NAME='Fortaleza'), (select
MAK_CNPJ from MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'));
INSERT INTO UMFASST (UMF_STL_ID, UMF_MAK_CNPJ) VALUES ((select
STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID where

```

```

STL_NAME='Porto das Dunas' and STT_NAME='Aquiraz')), (select MAK_CNPJ
from MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'));
INSERT INTO UMFASST (UMF_STL_ID, UMF_MAK_CNPJ) VALUES ((select
STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID where
STL_NAME='Coite' and STT_NAME='Eusebio'), (select MAK_CNPJ from
MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'));

INSERT INTO IMMOBILIE (IMM_PREIS, IMM_FLAECH, IMM_ZIMMERQUANTITAET,
IMM_EGR_PARKPLATZ, IMM_SONNENSTUNDEN, IMM_STRASSE, IMM_ANZAHL,
IMM_ERGARNZUNG, IMM_MAK_CNPJ, IMM_STL_ID) VALUES (1500000.00,
'400m2', 4, true, 'Manhã', 'Avenida Padre Antonio Tomas', 90,
'AP18000', (select MAK_CNPJ from MAKLERBUERO where
MAK_CNPJ='03.847.655/0001-98'), (select STL_ID from STADTTEIL inner
join STADT on STL_STT_ID=STT_ID where STL_NAME='Aldeota' and
STT_NAME='Fortaleza'));
INSERT INTO IMMOBILIE (IMM_PREIS, IMM_FLAECH, IMM_ZIMMERQUANTITAET,
IMM_EGR_PARKPLATZ, IMM_SONNENSTUNDEN, IMM_STRASSE, IMM_ANZAHL,
IMM_ERGARNZUNG, IMM_MAK_CNPJ, IMM_STL_ID) VALUES (950000.00,
'250m2', 2, true, 'Manhã', 'Rua Dep Moreira Rocha', 1903, 'AP402',
(select MAK_CNPJ from MAKLERBUERO where MAK_CNPJ='03.847.655/0001-
98'), (select STL_ID from STADTTEIL inner join STADT on
STL_STT_ID=STT_ID where STL_NAME='Meireles' and
STT_NAME='Fortaleza'));
INSERT INTO IMMOBILIE (IMM_PREIS, IMM_FLAECH, IMM_ZIMMERQUANTITAET,
IMM_EGR_PARKPLATZ, IMM_SONNENSTUNDEN, IMM_STRASSE, IMM_ANZAHL,
IMM_ERGARNZUNG, IMM_MAK_CNPJ, IMM_STL_ID) VALUES (2000000.00,
'190m2', 2, true, 'Manhã', 'Av Beira Mar', 1387, 'AP1203', (select
MAK_CNPJ from MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'),
(select STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID
where STL_NAME='Praia de Iracema' and STT_NAME='Fortaleza'));
INSERT INTO IMMOBILIE (IMM_PREIS, IMM_FLAECH, IMM_ZIMMERQUANTITAET,
IMM_EGR_PARKPLATZ, IMM_SONNENSTUNDEN, IMM_STRASSE, IMM_ANZAHL,
IMM_ERGARNZUNG, IMM_MAK_CNPJ, IMM_STL_ID) VALUES (700000.00, '90m2',
1, true, 'Manhã', 'Av dos Golfinhos', 200, 'BL2 AP303', (select
MAK_CNPJ from MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'),
(select STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID
where STL_NAME='Porto das Dunas' and STT_NAME='Aquiraz'));
INSERT INTO IMMOBILIE (IMM_PREIS, IMM_FLAECH, IMM_ZIMMERQUANTITAET,
IMM_EGR_PARKPLATZ, IMM_SONNENSTUNDEN, IMM_STRASSE, IMM_ANZAHL,
IMM_ERGARNZUNG, IMM_MAK_CNPJ, IMM_STL_ID) VALUES (190000.00,
'100m2', 2, true, 'Tarde', 'Estrada Jaboti', 889, 'Casa 1', (select
MAK_CNPJ from MAKLERBUERO where MAK_CNPJ='03.847.655/0001-98'),
(select STL_ID from STADTTEIL inner join STADT on STL_STT_ID=STT_ID
where STL_NAME='Coite' and STT_NAME='Eusebio'));

```

**Figura 38 - Modelo físico da Imobiliária Alemã**

## APÊNDICE 4 – CASOS DE USO E DIAGRAMAS DE SEQUÊNCIA

---

Este apêndice apresenta a especificação do caso de uso “Pesquisar imóvel” da aplicação “Submeter proposta de compra” a qual invoca os serviços identificados, analisados, projetados e implementados, além do diagrama de sequência do mesmo apresentado na Figura 39.

O caso de uso foi identificado a partir do modelo de processos descrito no Apêndice 1. Ele foi descrito de acordo com o formato de descrição de casos de uso apresentado por Bezerra [2006], considerando as informações principais para descrição dos casos de uso e incluindo extensão para explicitar quais atividades do modelo de processos são representadas no caso de uso. Por exemplo, o caso de uso “Pesquisar imóvel” representa as ações executadas nas atividades “Pesquisar imóvel”, “Analisar critérios de pesquisa”, “Obter imóveis disponíveis”, “Obter imóveis disponíveis em parceiras”, “Escolher imóvel” e “Submeter proposta de compra”.

### 1. Caso de uso: Pesquisar imóvel

**Relação com o modelo de processos:** Atividades “Pesquisar imóvel”, “Analisar critérios de pesquisa”, “Obter imóveis disponíveis”, “Obter imóveis disponíveis em parceiras”, “Escolher imóvel” e “Submeter proposta de compra”.

**Ator:** Cliente

**Ator Secundário:** Imobiliária parceira

**Tipo:** Primário e Essencial

**Descrição:** O cliente entra com os critérios de pesquisa para consultar os imóveis desejados. De acordo com estes critérios é feita uma busca na imobiliária Online e nas imobiliárias parceiras pelos imóveis que atendem aos critérios de pesquisa. As imobiliárias parceiras consultadas são apenas as que possuem imóveis na sua área de abrangência. A área de abrangência definida é o bairro que a imobiliária possui imóveis e disponibilizado para a imobiliária Online. Os imóveis

recuperados são apresentados para o cliente. Caso o cliente se interesse por algum imóvel ele pode submeter à proposta de compra, que será analisada pela imobiliária que está vendendo o imóvel.

**Pré-condições:** Não há

**Pós-condições:** Proposta de compra cadastrada

**Início:** O caso de uso inicia quando o Cliente deseja consultar os imóveis disponíveis.

**Fluxo principal:**

1. O Cliente inicia a busca informando o seu CPF e os critérios de pesquisa:
  - Bairro
  - Preço mínimo e/ou Preço máximo
  - Número de quartos
  - Se possui vaga de garagem
  - Se é sol da manhã e/ou da tarde
2. O Sistema retorna todos os imóveis da imobiliária Online e imobiliárias parceiras que atendem aos critérios de pesquisa, contendo as seguintes informações:
  - Nome da imobiliária
  - Preço
  - Área
  - Número de quartos
  - Se possui vaga de garagem
  - Se é sol da manhã e/ou da tarde
  - Endereço do imóvel: logradouro, número, complemento, bairro, cidade e estado
  - Opção de exibir detalhes do imóvel e Submeter proposta de compra
3. O Cliente escolhe um imóvel qual imóvel deseja visualizar.
4. O Sistema exibe as informações do imóvel, contendo:
  - Imóvel:
    - Área
    - Preço
    - Número de quartos
    - Se possui vaga de garagem

- Se é sol da manhã e/ou da tarde
    - Outros detalhes
    - Logradouro
    - Número
    - Complemento
    - Bairro
    - Cidade
    - Estado
  - Imobiliária
    - CNPJ
    - Nome
    - Telefone 1
    - Telefone 2
    - Telefone 3
    - Logradouro
    - Número
    - Complemento
    - Bairro
    - Cidade
    - Estado
5. O Cliente seleciona a opção submeter proposta de compra.
  6. O Sistema solicita o preenchimento dos dados da proposta de compra:
    - Nome
    - CPF
    - Valor da proposta
    - Descrição
  7. O Cliente submete a proposta de compra e confirma os dados.
  8. O Sistema cadastra a proposta de compra do imóvel para o Cliente e encerra o caso de uso.

**Fluxo alternativo 2.1:** Não existem imóveis para os critérios informados

1. O Sistema exibe mensagem informando que não há imóvel que atenda aos critérios de pesquisa.
2. Encerrar caso de uso.

**Fluxo alternativo 3.1:** O Cliente seleciona a opção Submeter proposta de compra

1. Ir para o passo 6 do caso de uso.

**Fluxo alternativo 6.1:** O imóvel não está mais disponível para venda.

1. O Sistema exibe mensagem informando que o imóvel não está mais disponível e remove o imóvel da lista de imóveis para os critérios de pesquisa informados.
2. Ir para passo 2 do caso de uso.

## **2. Caso de uso: Pesquisar imóvel**

O diagrama de sequência, apresentado na Figura 39, mostra a chamada de funções e invocação dos serviços implementados. Ele se inicia com o cliente clicando em um link para iniciar a pesquisa por imóveis. Este link foi nomeado como “Pesquisar imóveis” sendo um link disponibilizado em uma página JSP. Quando o cliente clica neste link, é chamada a função `antesPesquisar()` de um Servlet disponibilizado no servidor de aplicação do Portal. Este função tem a responsabilidade de invocar os serviços que retornam a abrangência das imobiliárias (OnLine, Alemã e Inglesa) com o objetivo de preencher a lista de bairros disponíveis para pesquisa. Esta página é exibida para o cliente juntamente com os outros campos que ele pode preencher na busca. O cliente então preenche os campos de pesquisa e clica no botão Pesquisar no qual submete a busca. Neste caso, nomeamos esta opção no diagrama como “Submeter pesquisa”. Ao executar esta opção da página JSP, a função `pesquisar()` do Servlet do Portal invoca o serviço de `registrarPesquisa()` e o(s) serviço(s) para listar os imóveis da(s) imobiliária(s) que possuem abrangência no(s) bairro(s) pesquisado(s). Neste caso, os métodos “`listarOnline()`”, “`listarAlema()`” e “`listarInglesa()`”. Quando os resultados são recebidos, o método `pesquisar()` chama o método `consolidarBusca()` que realiza a consolidação propriamente dita das listagens em única listagem. Esta lista única corresponde à lista dos imóveis que atendem aos critérios de pesquisa. Ela é apresentada em uma página para o cliente. Por fim, o cliente pode submeter uma proposta de compra para um imóvel (através da escolha do imóvel e preenchimento das informações da proposta). No diagrama, essa opção foi nomeada como “Submeter proposta”, no qual o cliente clica no botão Submeter executando o método “`submeterProposta()`” da página JSP. Este link invoca a função

`submeterProposta()` do Servlet que chama o serviço `registrarProposta()` para efetivar o cadastro da proposta propriamente dita.



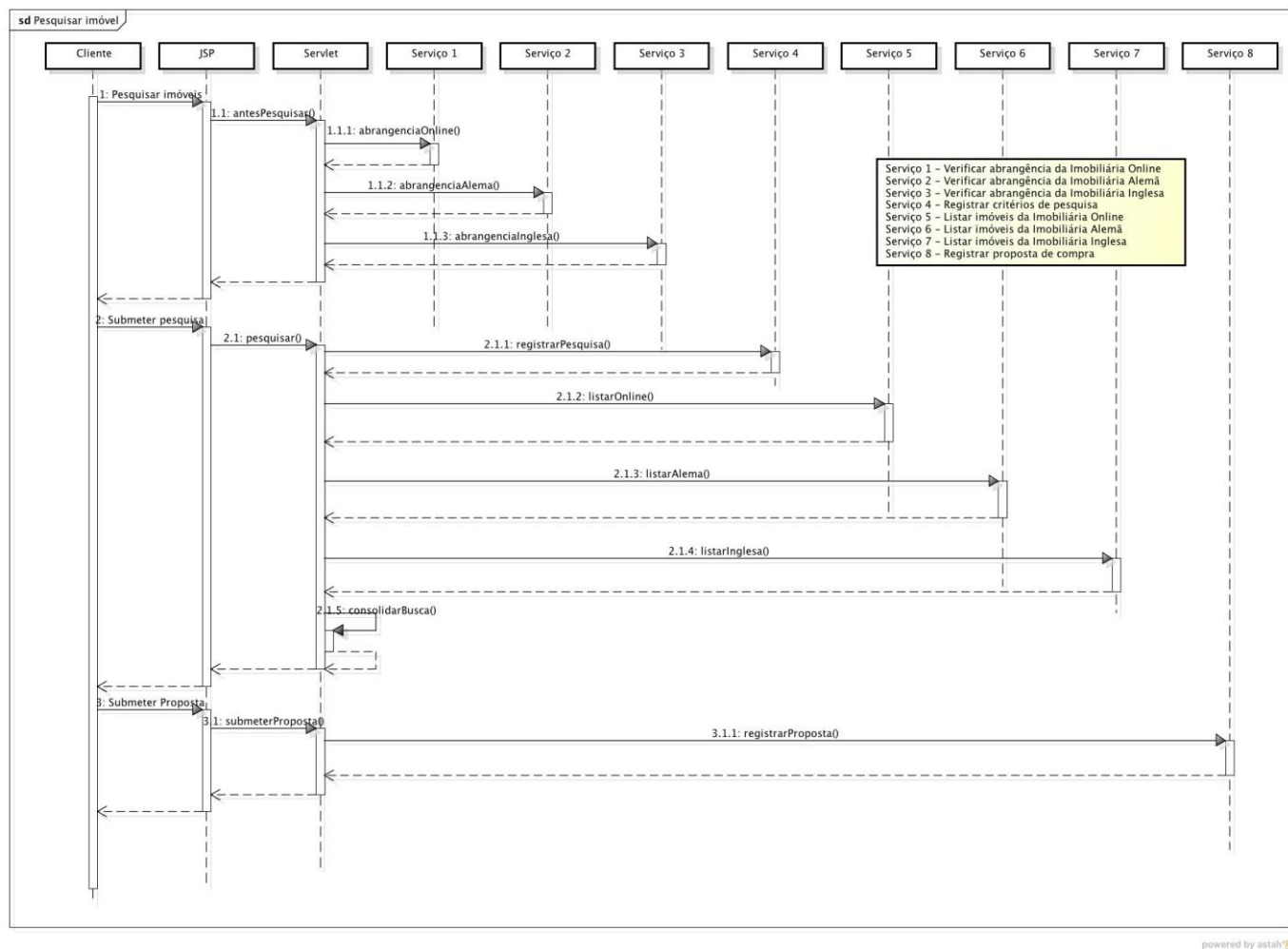


Figura 39 - Diagrama de sequência

## APÊNDICE 5 – DIAGRAMAS DE CLASSE

Este apêndice tem como objetivo apresentar os diagramas de classe de todos os serviços implementados e explicitados na seção 4.5.

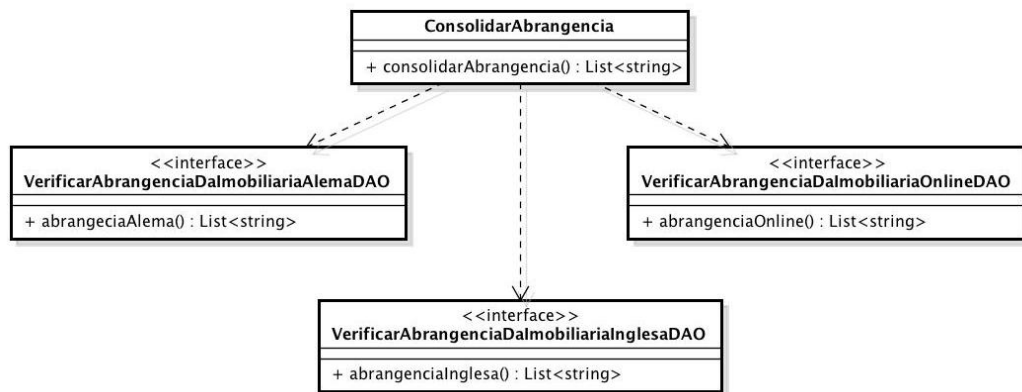


Figura 40 - Diagrama de classe do serviço Consolidar Abrangência

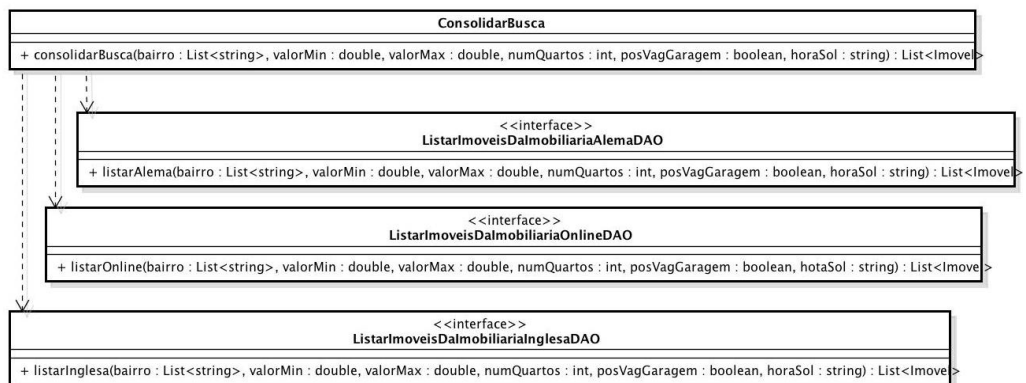


Figura 41 - Diagrama de classe do serviço Consolidar Busca

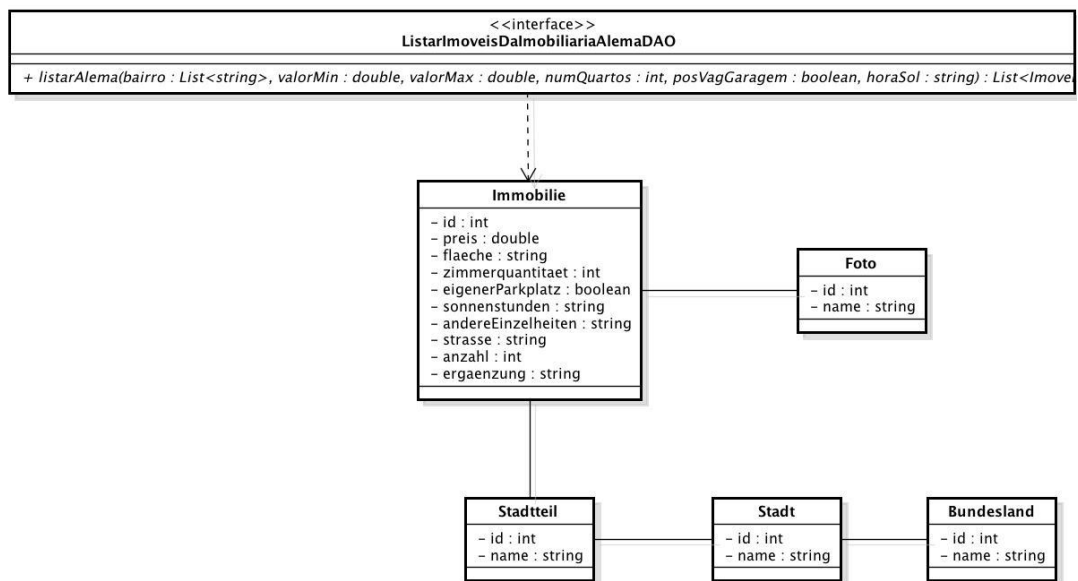


Figura 42 - Diagrama de classe do serviço Listar imóveis da imobiliária alemã

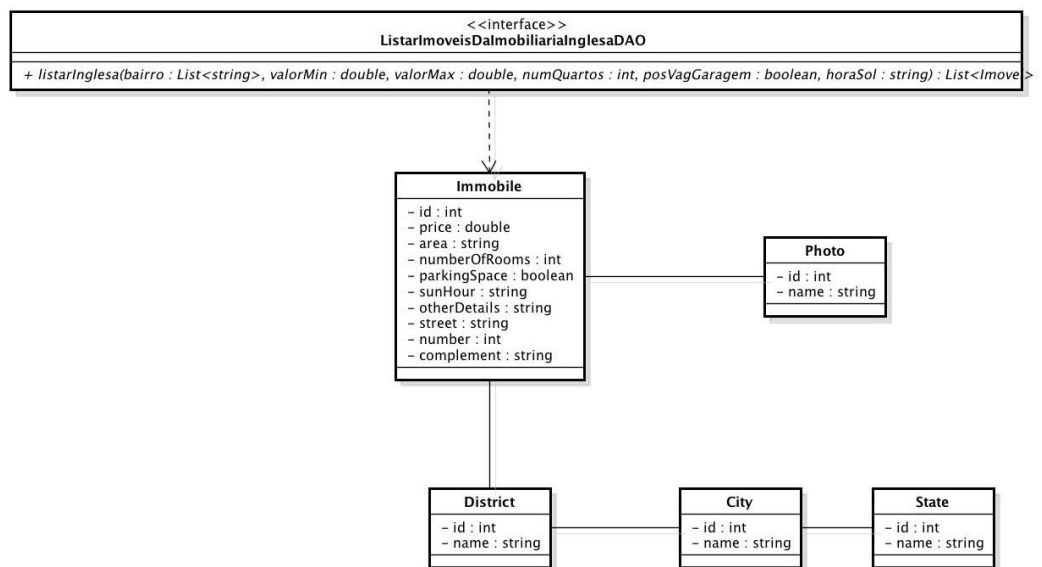


Figura 43 - Diagrama de classe do serviço Listar imóveis da imobiliária inglesa

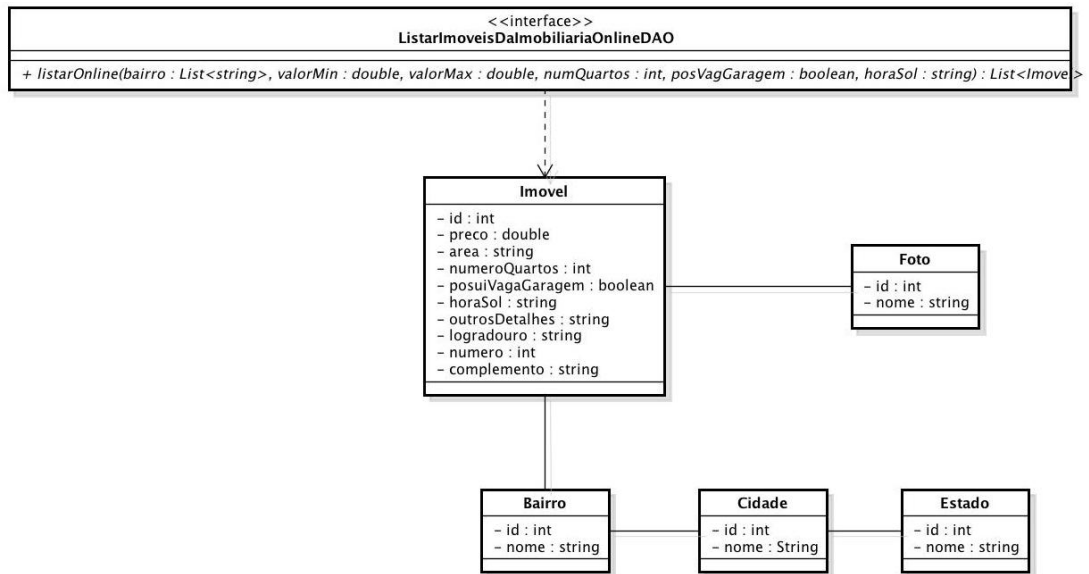


Figura 44 - Diagrama de classe do serviço Listar imóveis da imobiliária online

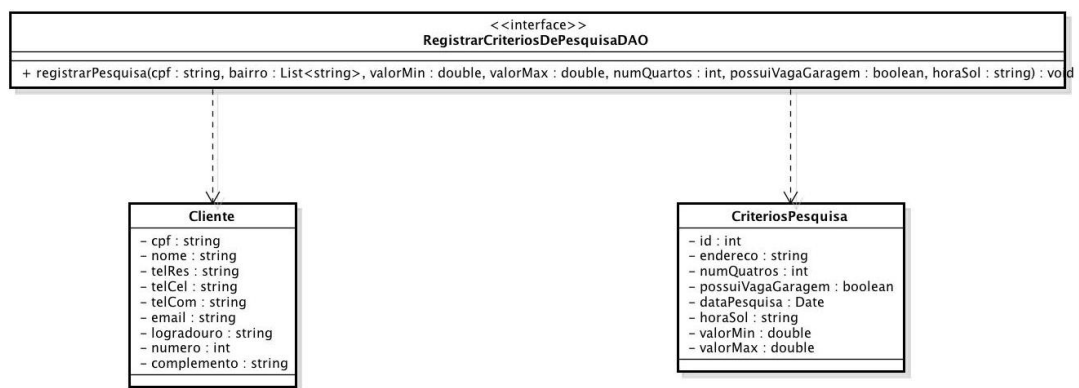


Figura 45 - Diagrama de classe do serviço Registrar critérios de pesquisa

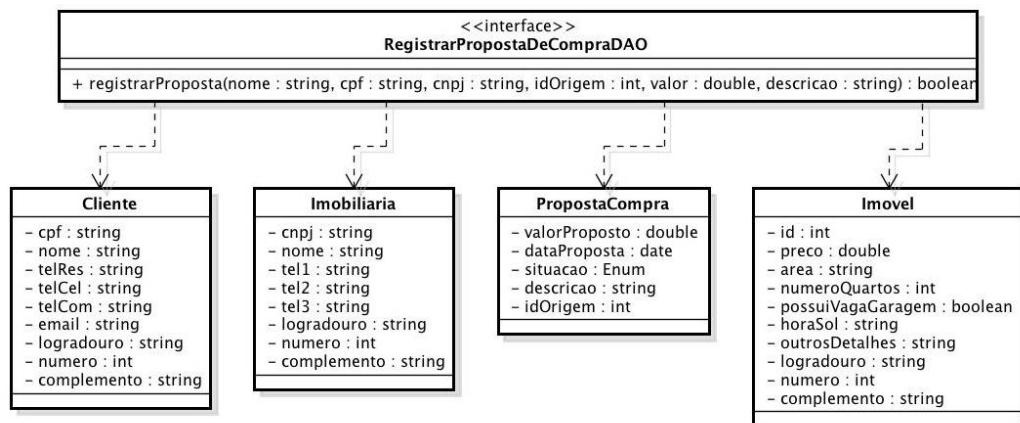


Figura 46 - Diagrama de classe do serviço Registrar proposta de compra

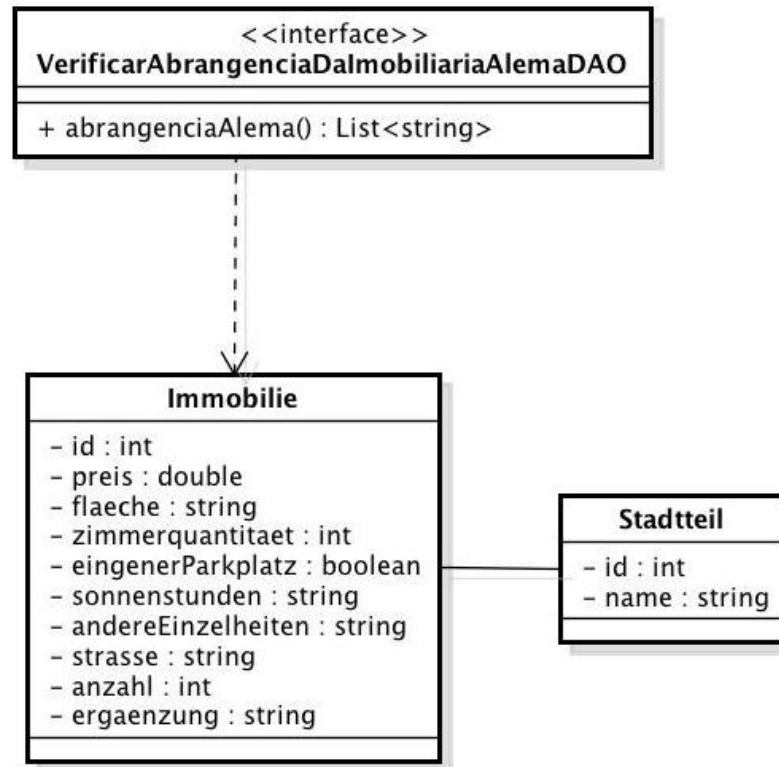


Figura 47 - Diagrama de classe do serviço Verificar abrangência da imobiliária alemã

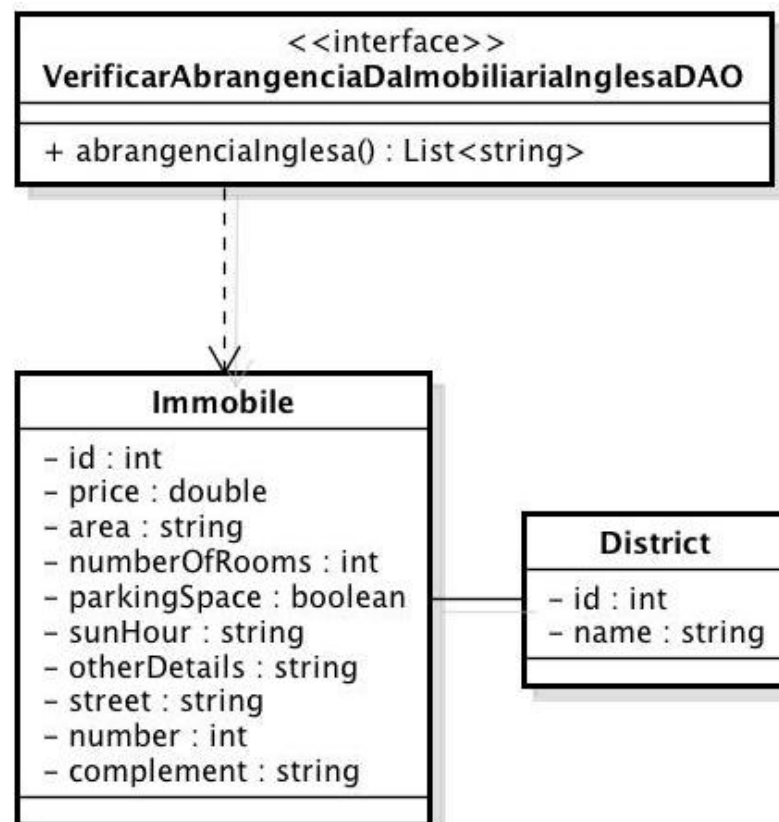


Figura 48 - Diagrama de classe do serviço Verificar abrangência da imobiliária inglesa

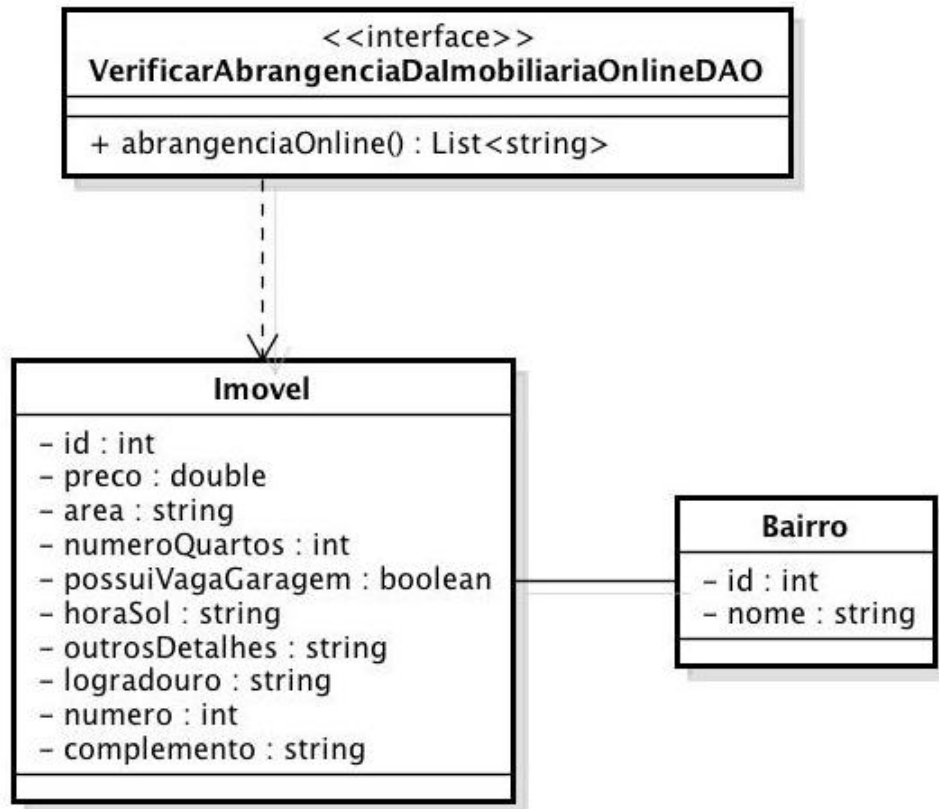


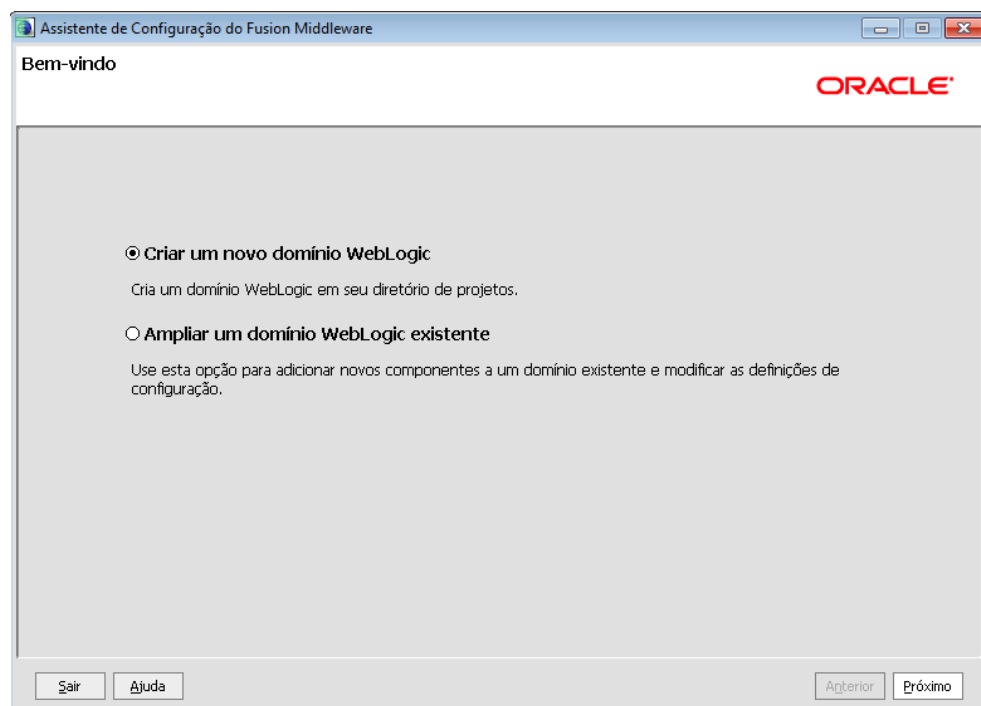
Figura 49 - Diagrama de classe do serviço Verificar abrangência da imobiliária online

## APÊNDICE 6 – CRIANDO UM NOVO DOMÍNIO NO ORACLE SERVICE BUS COM O ASSISTENTE DE CONFIGURAÇÃO

---

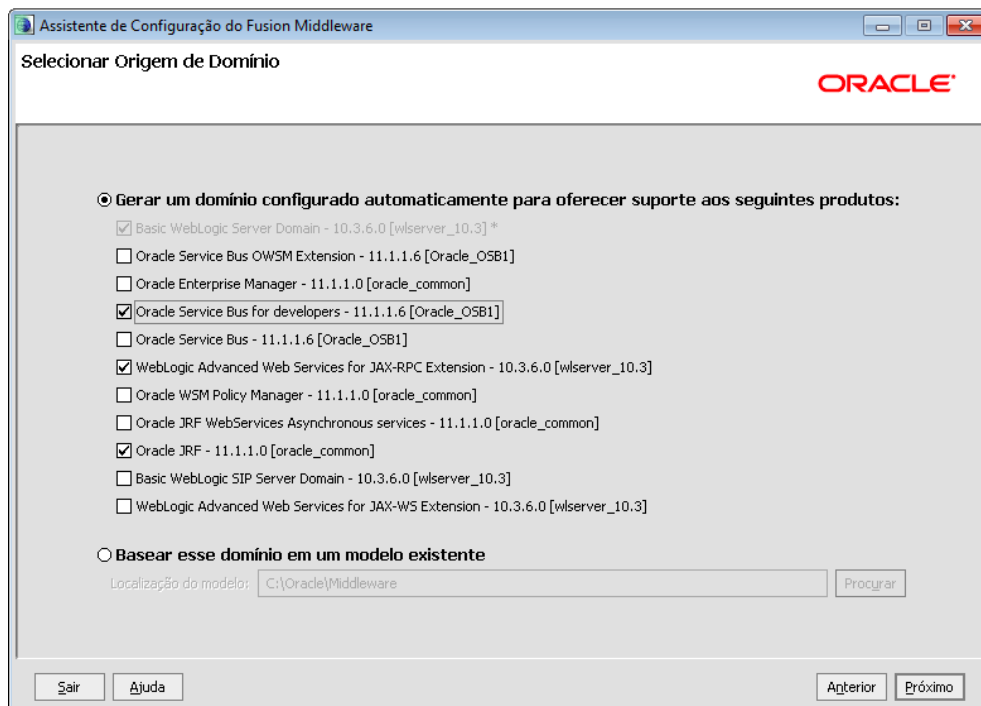
Este apêndice tem como objetivo apresentar um passo a passo da criação de um domínio no qual os serviços do projeto serão publicados.

Após instalar o Oracle Service Bus e seus pré-requisitos inicie o assistente de configuração a partir do menu iniciar. Logo que o assistente é iniciado uma tela semelhante a da Figura 50 é exibida e deve-se selecionar a opção *Criar um novo domínio WebLogic* e clicar em *Próximo*.



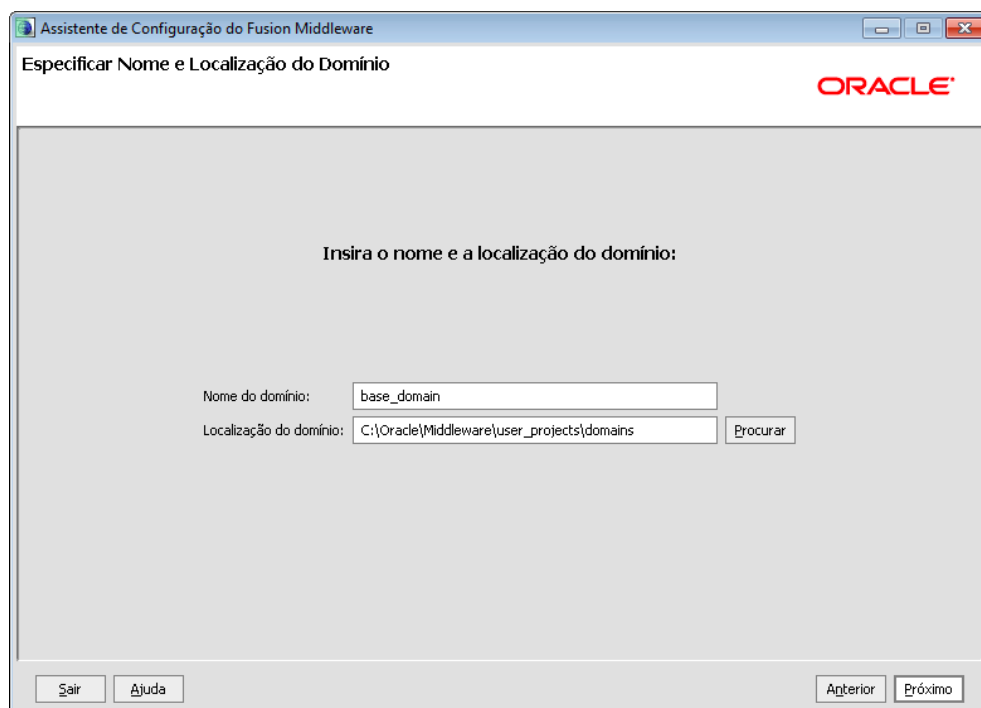
**Figura 50 - Tela inicial do assistente de configuração**

Em seguida selecionar a opção *Gerar um domínio configurado automaticamente para oferecer suporte aos seguintes produtos* e selecionar as subopções *Oracle Service Bus for developers – 11.1.1.6*; *WebLogicAdvanced Web Services for JAX-RPC Extension – 10.3.6.0*; e, *Oracle JRF – 11.1.1.0* (Figura 51).



**Figura 51 - Selecionando origem de domínio no assistente de configuração**

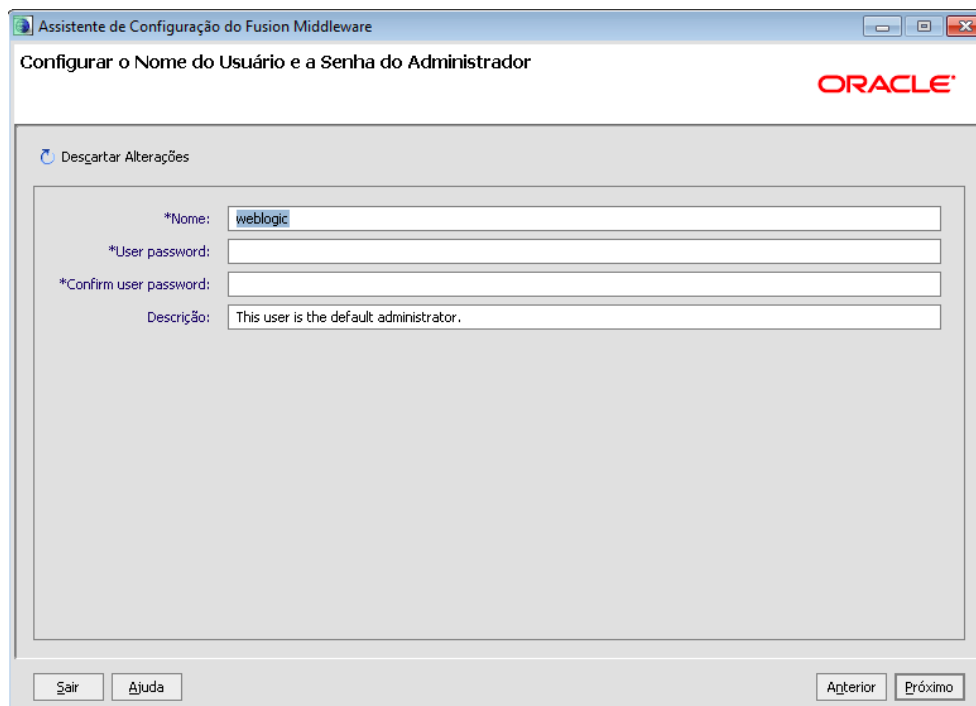
Na tela seguinte devem-se preencher os campos *Nome do domínio* e definir a *Localização do domínio* como apresentado na Figura 52.



**Figura 52 - Especificando nome e localização do domínio**

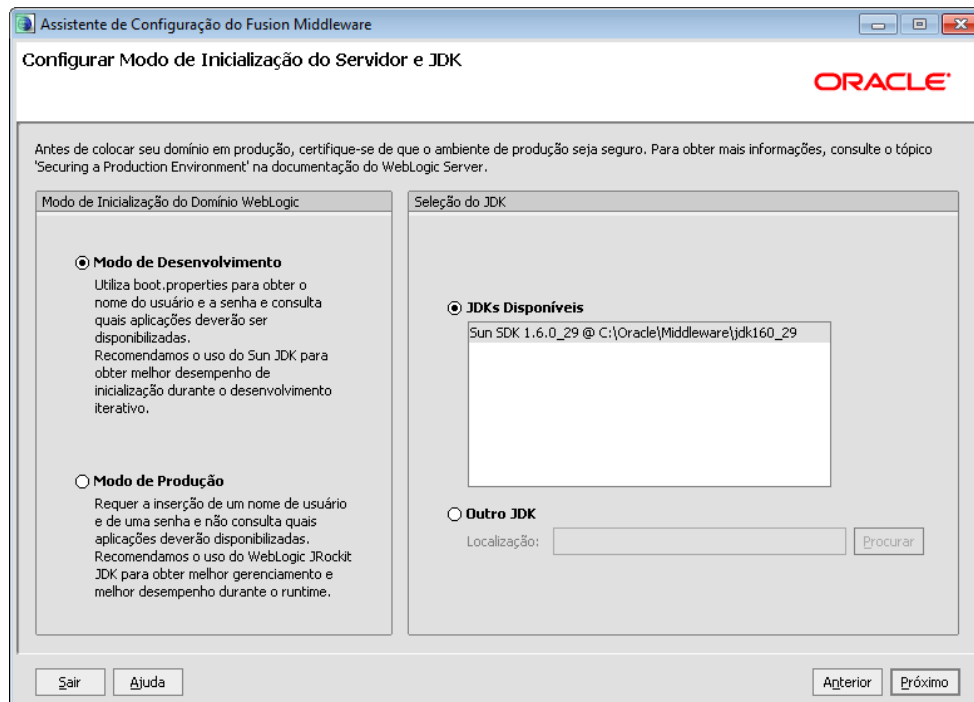


Em seguida é possível configurar o nome de usuário e senha do administrador (Figura 53), para isso é necessário preencher os campos: *Nome*; *User password*; *Confirm user password* e *Descrição*.



**Figura 53 - Configurando o nome do usuário e senha do administrador**

Na tela seguinte é possível configurar o modo de inicialização do servidor e o JDK para o mesmo. Como o objetivo é publicar e disponibilizar os serviços no ESB é necessário selecionar o *Modo de Desenvolvimento* e selecionar o *JDK* disponível, como ilustrado na Figura 54.



**Figura 54 - Configurando o modo de inicialização do servidor e JDK**

Em seguida é possível configurar o esquema do componente do JDBC. Para atender aos requisitos deste trabalho, foi selecionada a configuração padrão, uma vez que o ESB será o mediador entre o consumidor e o provedor de serviços. Como os serviços foram desenvolvidos no NetBeans e disponibilizados no servidor de aplicação GlassFish, o ESB tem como função rotear as chamadas para os serviços provenientes dos consumidores para os serviços disponibilizados no Glassfish. No Glassfish, os drives necessários para os serviços se conectarem ao banco de dados foram configurados. Logo, selecionamos a opção *Próximo* na tela correspondente a Figura 55.

**Assistente de Configuração do Fusion Middleware**

**Configurar Esquema do Componente JDBC**

**Observação:** Altere somente os campos de entrada a seguir que deseja modificar e os valores serão aplicados a todas as linhas selecionadas.

Fornecedor:  DBMS/Serviço:

Driver:  Nome do Host:

Proprietário do Esquema:  Porta:

Senha de Esquema:

RAC configuration for component schemas:

☐ Convert to GridLink ☐ Convert to RAC multi data source ☐ Don't convert

	Esquema do Componente	DBMS/Serviço	Nome do Host	Porta	Proprietário do Esq...	Senha de Esquema
<input type="checkbox"/>	OSB JMS Reporting Provider	osbexamples	localhost	1527	DEV_SOAINFRA	*****

Sair Ajuda Anterior Próximo

**Figura 55 - Configurando esquema do componente JDBC**

Na tela seguinte, o assistente irá testar o esquema do componente, como apresentado na Figura 56, após o teste é possível clicar em *Próximo* para seguir com as configurações.

**Assistente de Configuração do Fusion Middleware**

**Testar Esquema do Componente**

Status	Esquema do Componente	URL para Conexão JDBC
<input checked="" type="checkbox"/>	OSB JMS Reporting Provider	jdbc:derby://localhost:1527/osbexamples;ServerName=localhost;databaseName=osbexamples

Selecionar Tudo Cancelar Seleção de Tudo Testar Conexões

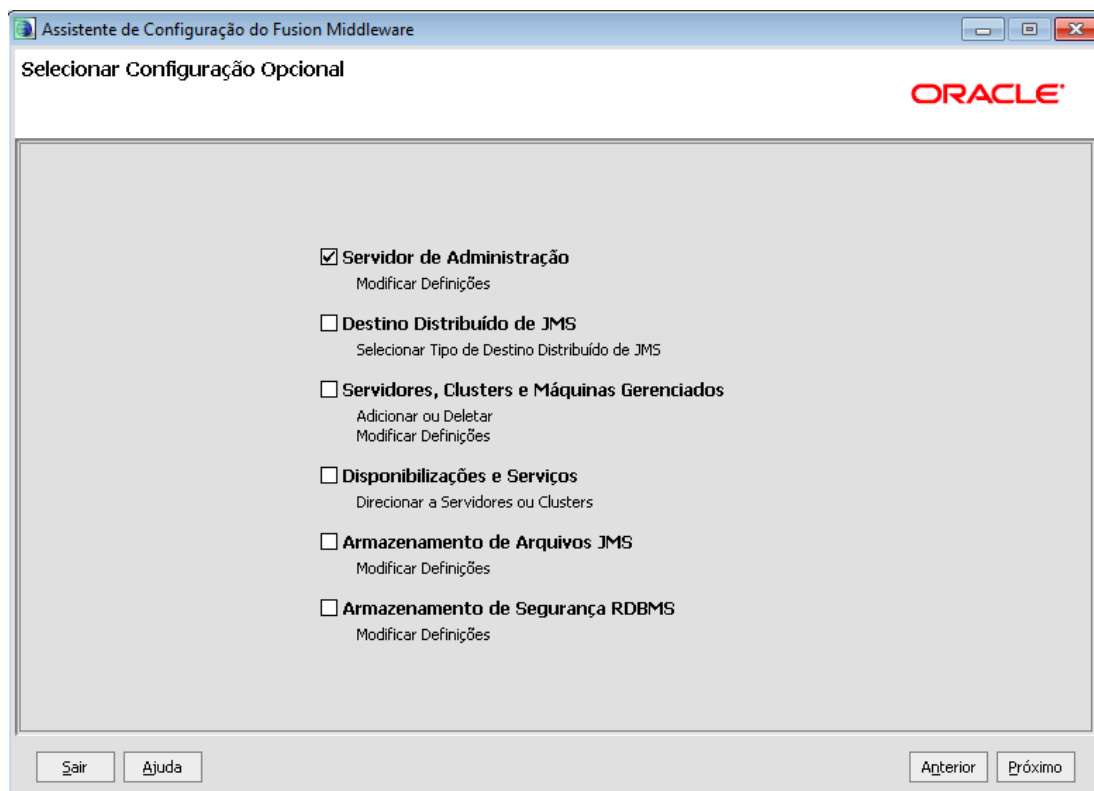
Log do Resultado da Conexão

Esquema do Componente=OSB JMS Reporting Provider  
 Driver=org.apache.derby.jdbc.ClientDriver  
 URL=jdbc:derby://localhost:1527/osbexamples;ServerName=localhost;databaseName=osbexamples  
 Usuário=DEV\_SOAINFRA  
 Senha=\*\*\*\*\*  
 Teste SQL=SELECT 1 FROM SYS.SYSTABLES  
 CFGFWK-20850: Teste Bem-Sucedido!

Sair Ajuda Anterior Próximo

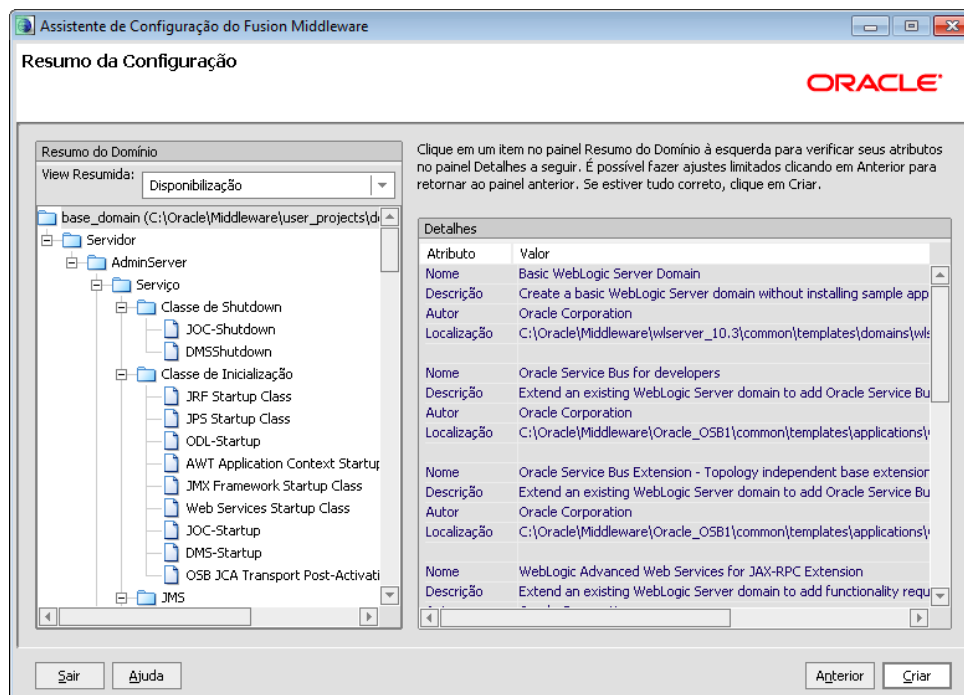
**Figura 56 - Teste do esquema do componente**

Em seguida é exibida uma tela para selecionar configurações opcionais, na qual foi selecionada a opção *Servidor de Administração*, pois os serviços serão publicados no ESB o qual fará o roteamento para os servidores onde os serviços estão realmente executando (Figura 57).



**Figura 57 - Selecionando configurações opcionais**

Por fim, será exibido um resumo das configurações efetuadas. Todos os dados devem ser confirmados e deve-se clicar na opção *Criar*, como apresentado na Figura 58. Ao clicar no botão *Criar* o assistente cria o domínio automaticamente informando ao final do processo se o mesmo foi bem sucedido.



**Figura 58 - Resumo das configurações**

## APÊNDICE 7 – IMPLEMENTAÇÕES

---

Este apêndice apresenta uma implementação de um serviço, classe de acesso a dados e de um método do servlet para exemplificação do código implementado. As implementações de todos os serviços, servlets e modelos físicos podem ser acessadas através do endereço <http://code.google.com/p/pf-aplicacao-metodo-bpm-soa/>.

Como exemplo de serviço utilizamos o “Verificar Abrangência da Imobiliária Online”, que é apresentado na Figura 59. Este serviço invoca o método da classe “VerificarAbrangenciaDaImobiliariaOnlineDAO” representado pela Figura 60.

```
package br.unirio.pf.service;

import br.unirio.pf.dao.VerificarAbrangenciaDaImobiliariaOnlineDAO;
import java.util.List;
import javax.jws.WebService;
import javax.jws.WebMethod;

@WebService(serviceName = "VerificarAbrangenciaDaImobiliariaOnline")
public class VerificarAbrangenciaDaImobiliariaOnline {

    @WebMethod
    public List<String> abrangenciaOnline() throws Exception {
        VerificarAbrangenciaDaImobiliariaOnlineDAO dao = new
        VerificarAbrangenciaDaImobiliariaOnlineDAO();
        List<String> abrangenciaOnline = dao.abrangenciaOnline();

        return abrangenciaOnline;
    }
}
```

**Figura 59 - Serviço Verificar Abrangência da Imobiliária Online**

```
package br.unirio.pf.dao;

import br.unirio.pf.entidade.Conexao;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;

public class VerificarAbrangenciaDaImobiliariaOnlineDAO {

    private static VerificarAbrangenciaDaImobiliariaOnlineDAO
    instancia;
    private static Conexao conexaoInstancias;
    protected ConexaoImbOnline dao = new ConexaoImbOnline();
    Connection conexao;
```

```

    public static VerificarAbrangenciaDaImobiliariaOnlineDAO
getInstance(){
    if(instancia == null){
        instancia = new
VerificarAbrangenciaDaImobiliariaOnlineDAO();
        conexaoInstancias = new
Conexao("jdbc:postgresql://localhost:5432/BD_Online","postgres","123
45678");
    }
    return instancia;
}

public List<String> abrangenciaOnline() throws Exception{
    List<String> listBairro = new ArrayList<String>();
    String bairro = null;
    try{
        conexao = dao.abrirConexao(false, conexaoInstancias);
        Statement stmt = conexao.createStatement();
        ResultSet rs = null;

        StringBuilder sql = new StringBuilder();
        sql.append("select distinct bro_nome from bairro inner
join imovel on bro_id=imv_bro_id inner join possui on
pos_imv_id=imv_id");
        rs = stmt.executeQuery(sql.toString());

        while(rs.next()){
            bairro = rs.getString("bro_nome");
            listBairro.add(bairro);
        }
    }finally{
        if (!conexao.isClosed()){
            ConexaoImbOnline.fecharConexao();
        }
    }
    return listBairro;
}
}

```

**Figura 60 – DAO do serviço Verificar Abrangência da Imobiliária Online**

Como exemplo de método do servlet, é apresentado o método “Submeter Proposta”, referente à página de submissão de propostas (Figura 24), aqui representado pela Figura 61. Neste método, os valores preenchidos pelo usuário na página de submissão da proposta são capturados em variáveis locais (nome, cpf, valorProposta e descricao) e o método “Registrar Proposta” da classe Cliente é invocado. A classe “Cliente” contém as referências, implementadas via métodos, para todos os serviços utilizados pela aplicação consumidora dos serviços.

```

protected void submeterProposta(HttpServletRequest request, HttpServletResponse response) throws Exception{
    String nome = String.valueOf(request.getParameter("nome"));
    String cpf = String.valueOf(request.getParameter("CPF"));
    double valorProposta =
Double.valueOf(request.getParameter("valorProposta"));
    String descricao =
String.valueOf(request.getParameter("descricao"));

    boolean proposta = Clientes.registrarProposta(nome, cpf,
cnpj, id, valorProposta, descricao);

    if(!proposta){
        String mensagem = "Proposta não cadastrada, tente
novamente!";
        request.setAttribute("mensagem", mensagem);
    }
    else{
        String mensagem = "Proposta cadastrada com sucesso!";
        request.setAttribute("mensagem", mensagem);
    }

    RequestDispatcher rd =
request.getRequestDispatcher("/pages/dadosProposta.jsp");
    rd.forward(request, response);
}

```

**Figura 61 - Método Submeter proposta**