

*Renato Dos Santos*

*Gabriel Neto Siqueira*

*Rodrigo Yoshida Lombezzi*

*Felipe Villela De Souza*

**SISTEMA DE CONTROLE DE IMOBILIÁRIA**

**Projeto Integrador**

Disciplinas Envolvidas: Engenharia de Software, Tópicos Avançado de Banco de Dados Relacional e não Relacional, Técnicas Avançadas de Programação, Técnicas Avançadas de Programação Web e Mobile, Gestão Ágil de Projetos de Software.

Jales

2023

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc144809989)

[2 LEVANTAMENTO DE REQUISITOS DE SOFTWARE 4](#_Toc144809990)

[2.1 Descrição dos objetivos do sistema 4](#_Toc144809991)

[2.2 Descrição do sistema atual 4](#_Toc144809992)

[2.3 Descrição dos principais problemas 4](#_Toc144809993)

[2.4 Descrição dos requisitos funcionais 4](#_Toc144809994)

[2.5 Descrição dos requisitos não funcionais 4](#_Toc144809995)

[3 VISÃO DE CASO DE USO – UML 5](#_Toc144809996)

[3.1 Diagrama de Classes 5](#_Toc144809997)

[3.2 Dicionário de classes 5](#_Toc144809998)

[3.3 Definição dos Atores 5](#_Toc144809999)

[3.4 Lista de Casos de Uso 5](#_Toc144810000)

[3.4. Diagrama de Casos de Uso 5](#_Toc144810001)

[3.5. Diagrama de Casos de uso individuais 6](#_Toc144810002)

[3.6. Diagrama de Sequência 7](#_Toc144810003)

[3.7. Diagrama de Comunicação 7](#_Toc144810004)

[3.8. Diagrama de Atividade 7](#_Toc144810005)

[4 DEFINIÇÃO DA INTERFACE COM O USUÁRIO (UX) (3º semstre) 8](#_Toc144810006)

[4.1 Descrição de cenário 8](#_Toc144810007)

[4.2 Descrição de personas 8](#_Toc144810008)

[4.3 Esboços de tela (wireframes) 8](#_Toc144810009)

[4.4 Protótipos de tela 8](#_Toc144810010)

[5 BANCO DE DADOS 9](#_Toc144810011)

[5.1 Modelo Entidade Relacionamento 9](#_Toc144810012)

[5.2 Script das tabelas 9](#_Toc144810013)

[6 ARQUITETURA DE SOFTWARE 10](#_Toc144810014)

[6.1 Arquitetura de desenvolvimento 10](#_Toc144810015)

[6.2 Telas do sistema 10](#_Toc144810016)

[7 CONCLUSÃO 11](#_Toc144810017)

[8 REFERÊNCIAS 12](#_Toc144810018)

# INTRODUÇÃO

A crescente complexidade do mercado imobiliário exige uma gestão eficiente e estratégica por parte das imobiliárias para garantir competitividade e excelência no serviço oferecido aos clientes. Neste contexto, a implementação de um sistema especializado torna-se não apenas uma vantagem, mas uma necessidade premente. O projeto em questão visa preencher essa lacuna ao desenvolver um sistema de gestão imobiliária abrangente e sofisticado, destinado a otimizar todas as etapas envolvidas na compra, venda, locação e administração de imóveis.

O sistema proposto não apenas simplifica e agiliza os processos internos da imobiliária, mas também promove uma abordagem holística para a gestão de propriedades, com foco na organização eficiente de informações e na maximização da transparência para todas as partes envolvidas. Ao fornecer uma plataforma centralizada, o sistema permite o cadastro e gerenciamento detalhado de todas as propriedades sob a responsabilidade da imobiliária, incluindo aspectos físicos, legais e de disponibilidade.

Além disso, o sistema oferece uma solução robusta para o armazenamento e gerenciamento de contratos de locação, acordos de compra e venda, e outros documentos cruciais relacionados às transações imobiliárias. Essa funcionalidade não só simplifica o processo de documentação, mas também garante a conformidade legal e a rastreabilidade de prazos, mitigando potenciais riscos e aumentando a segurança jurídica para todas as partes envolvidas.

A integração de ferramentas de automação e geração de relatórios permite uma análise mais precisa e rápida do desempenho do negócio, facilitando a tomada de decisões informadas e estratégicas. Além disso, o sistema pode oferecer recursos avançados de marketing, como a divulgação automatizada de imóveis em diferentes plataformas online, ampliando a visibilidade e o alcance da imobiliária no mercado.

Com um foco claro na produtividade, organização e transparência, o sistema de gestão imobiliária aqui proposto não apenas atende às demandas operacionais das imobiliárias modernas, mas também eleva o padrão de serviço oferecido aos clientes e parceiros, promovendo eficiência, confiabilidade e excelência em todas as interações e transações.

# LEVANTAMENTO DE REQUISITOS DE SOFTWARE

## Descrição dos objetivos do sistema

O sistema aborda a área do setor imobiliário, e vem com o objetivo de conseguir facilitar todos os processos por parte da empresa, fazendo uma maior gestão da compra, venda e locação dos imóveis que passaram pela imobiliária, além de salvar seus registros e facilitar a organização e o gerenciamento de pagamentos.

## Descrição do sistema atual

O sistema atual da imobiliária consiste em cadastrar, atualizar e gerenciar as informações das propriedades contendo fotos, documentos e histórico de vendas ou aluguéis da propriedade. O sistema também conta com CRM integrado, que mantém um banco de dados de clientes completo e organiza as interações, compromissos e histórico de comunicação para uma boa gestão de relacionamento com o cliente. O sistema possui uma automação de marketing que cria campanhas direcionadas para avaliar o progresso do seu negócio, identificando tendências e tomar decisões informadas. Possui também integração com portais imobiliários que publica automaticamente listagens de propriedades em portais imobiliários para aumentar a visibilidade e atrair mais clientes.

## Descrição dos principais problemas

Armazenar dados dos imóveis vendidos e locados, gerenciar o pagamento de um imóvel vendido, a parcela de locação e a depender do contrato, quem irá pagar as taxas, se será pago pelo locatário acréscimo na parcela ou se será descontado do locador e trazer todos os registros do histórico dos imóveis.

## Descrição dos requisitos funcionais

Os requisitos funcionais de um sistema planejam suas operações e capacidades, sendo moldados pela natureza do software em construção, pelos usuários previstos e pela estratégia geral adotada pela empresa ao elaborar tais elementos. Ao serem expressos como requisitos dos usuários e gestores do sistema, é fundamental que esses elementos funcionais ofereçam uma descrição minuciosa das operações do sistema, suas entradas, saídas e como lidam com situações excepcionais.

**Cadastro:** O sistema deve fazer o cadastro de imóvel, cadastro de funcionário e cadastro de cliente;

**Controle imóvel:** O sistema deve fazer o controle dos imóveis de venda e locado, gerenciando o estado do imóvel se está vendido ou locado;

**Controle pagamento:** O sistema deve fazer o controle de pagamento de venda e de locação, na locação o sistema fará o controle de quem irá pagar as taxas, se será pago pelo locatário acréscimo na parcela ou se será descontado do locador.

## Descrição dos requisitos não funcionais

**Otimização do sistema:** O sistema deve ser capaz de lidar com grandes volumes de dados, processamento rápido de consultas e atualizações, garantindo tempos de resposta aceitáveis mesmo em picos de uso.

**Utilização do banco de dados:** Utilização do banco de dados para conseguir guardar as informações de todos os usuários e clientes de uma maneira mais confiável e com a maior quantidade possível.

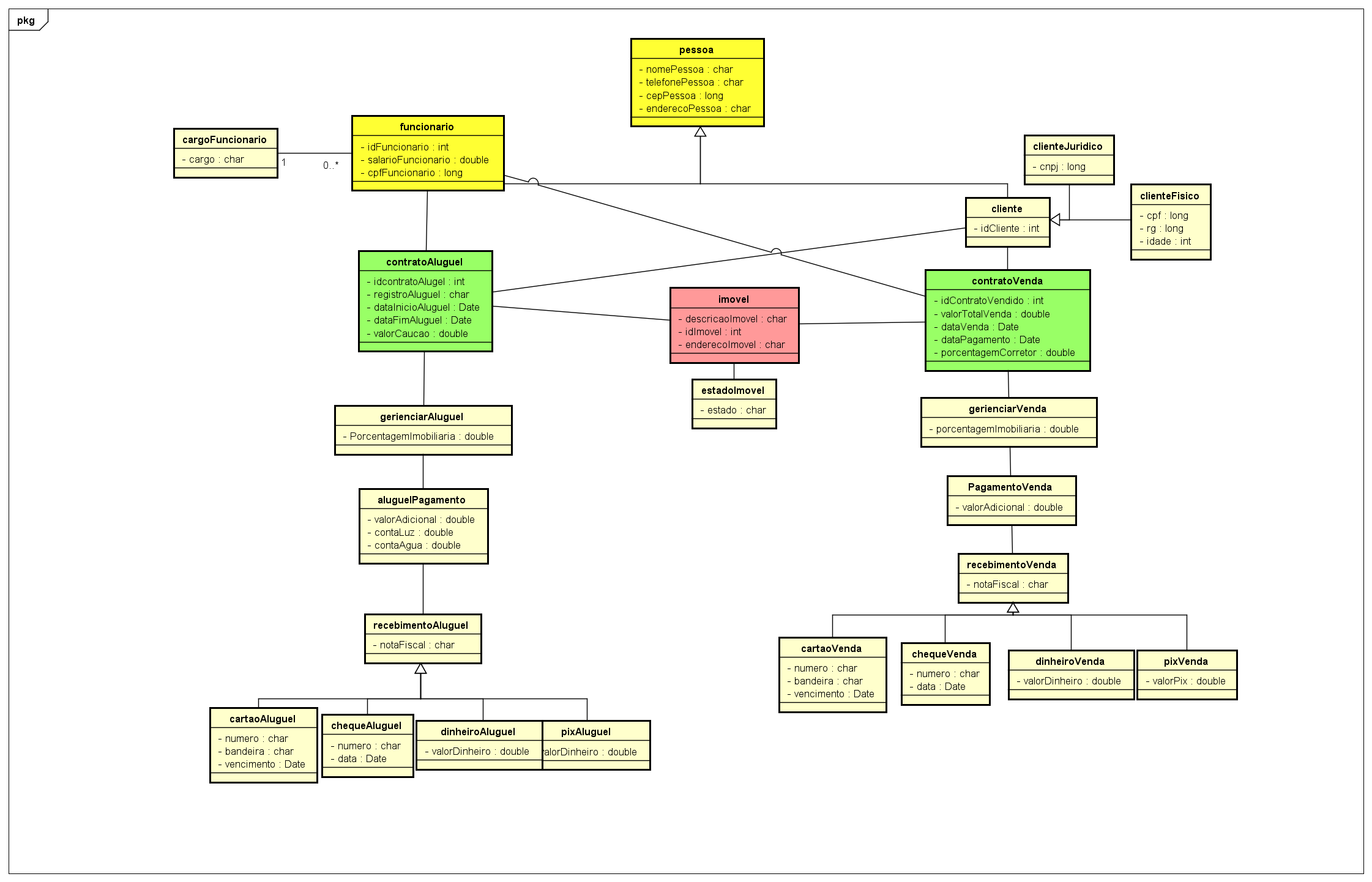
**Confiabilidade:** O sistema deve ser altamente confiável, minimizando o tempo de inatividade e garantindo a recuperação rápida em caso de falhas. Isso pode incluir redundância de servidores, backups regulares e mecanismos de monitoramento contínuo.

**Escalabilidade:** Deve ser capaz de se expandir para lidar com um aumento no número de usuários, transações e dados sem comprometer o desempenho. Isso pode envolver arquitetura modular, capacidade de adicionar novos servidores conforme necessário e design que suporte crescimento incremental.

**Disponibilidade:** O sistema deve estar disponível 24/7 para garantir que os agentes imobiliários possam acessar informações e realizar transações a qualquer momento.

# VISÃO DE CASO DE USO – UML

## 3.1 Diagrama de Classes



## 3.2 Dicionário de classes

Pessoa – A tabela pessoa tem como objetivo padronizar as informações que podem vir a ter em outras classes como, funcionário e cliente.

Tendo informações importantes para o reconhecimento do mesmo, como nome, telefone, endereço etc.

**Quadro 1** – Descrição Classe Pessoa

|  |  |  |
| --- | --- | --- |
| **Atributo** | **Tipo** | **Descrição** |
| idPessoa | Integer | Código único para identificação da pessoa. Esse campo é geralmente uma chave primária no banco de dados. |
| nomePessoa | Varchar | Nome completo da pessoa. Este campo armazena o nome e sobrenome da pessoa. Pode incluir letras, espaços e caracteres especiais permitidos em nomes. |
| telefonePessoa | Varchar | Número de telefone da pessoa. Este campo pode incluir o código de área e pode armazenar caracteres como parênteses, espaços e hifens para formatação. |
| cepPessoa | Varchar | Código de Endereçamento Postal (CEP) da pessoa. Este campo armazena o código postal que identifica a localização geográfica do endereço da pessoa. |
| enderecoPessoa | Varchar | Endereço completo da pessoa. Este campo inclui o nome da rua, número, complemento, bairro, cidade e estado. |
| idade | Integer | Idade da pessoa em anos completos. Este campo armazena um número inteiro representando quantos anos a pessoa tem. |

Funcionário - A tabela Funcionário é uma extensão da tabela Pessoa, e tem como objetivo armazenar informações adicionais sobre os funcionários da empresa. Utiliza atributos comuns da tabela Pessoa para garantir consistência e padronização, e inclui dados adicionais como salário, CPF e data de admissão, que são cruciais para a gestão dos recursos humanos.

Quadro 2 - Descrição Classe Funcionário

|  |  |  |
| --- | --- | --- |
| **Atributos** | **Tipo** | **Descrição** |
| |  | | --- | | idFuncionario | | Integer | Código único para identificação do funcionário. Este campo é geralmente uma chave primária. |
| CpfFuncionario | Varchar | Código de identificação associado à pessoa na tabela Pessoa. Este campo é uma chave estrangeira. |
| salarioFuncionario | Decimal | Salário do funcionário. Este campo armazena valores monetários. |
| dataAdmissao | Date | Data de admissão do funcionário na empresa. |

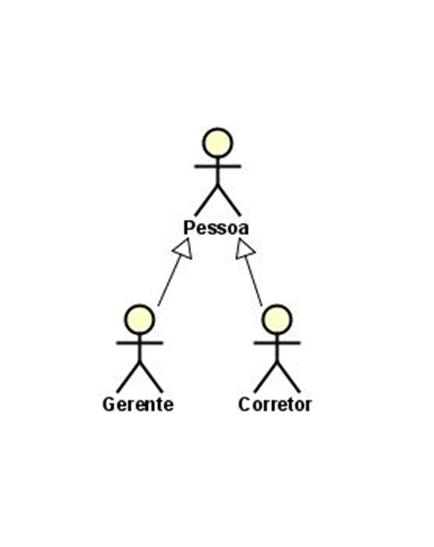
Cliente - A tabela Cliente é uma extensão da tabela Pessoa e tem como objetivo armazenar informações adicionais específicas sobre os clientes da empresa. Utiliza atributos comuns da tabela Pessoa para garantir consistência e padronização, e inclui dados adicionais como a data de cadastro, que são cruciais para a gestão de relacionamento com o cliente.

**Quadro -** Descrição classe Cliente

|  |  |  |
| --- | --- | --- |
| **Atributos** | **Tipo** | **Descrição** |
| idCliente | Integer | Código único para identificação do cliente. Este campo é geralmente uma chave primária. |
| dataCadastro | Date | Data de cadastro do cliente no sistema. |

Fonte: Elaborado pelos autores

## 3.3 Definição dos Atores



Ator Pessoa: Vai interagir com todas as funcionalidades do sistema e terá todas as permissões do corretor e do Gerente que vão herdar as funcionalidades do ator pessoa.

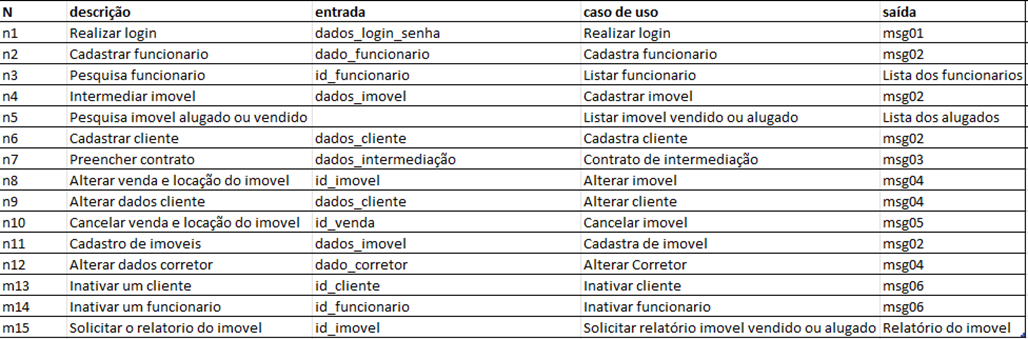
Pessoa gerente: interage com o sistema com todas as permissões

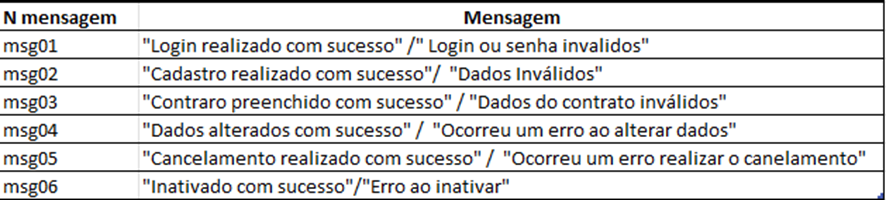
E funcionalidades do corretor e mais cadastro do funcionário, alterar dado funcionário, consultar funcionário e inativar funcionário.

Pessoa corretor: Interage com o sistema com as

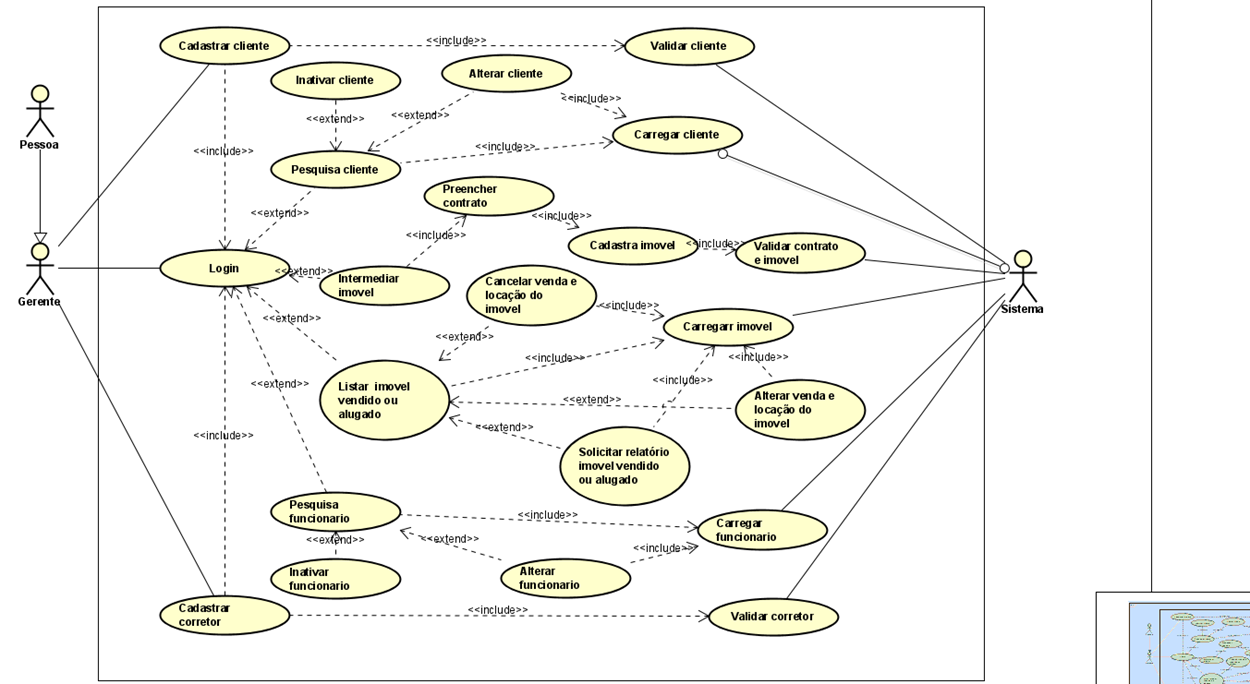
Funcionalidades: Realizar login, cadastrar cliente, cadastra imóvel, alterar venda e locação, alterar cliente, cancelar e venda e locação, preencher contrato, consultar clientes, inativar cliente e solicitar registros de imóvel.

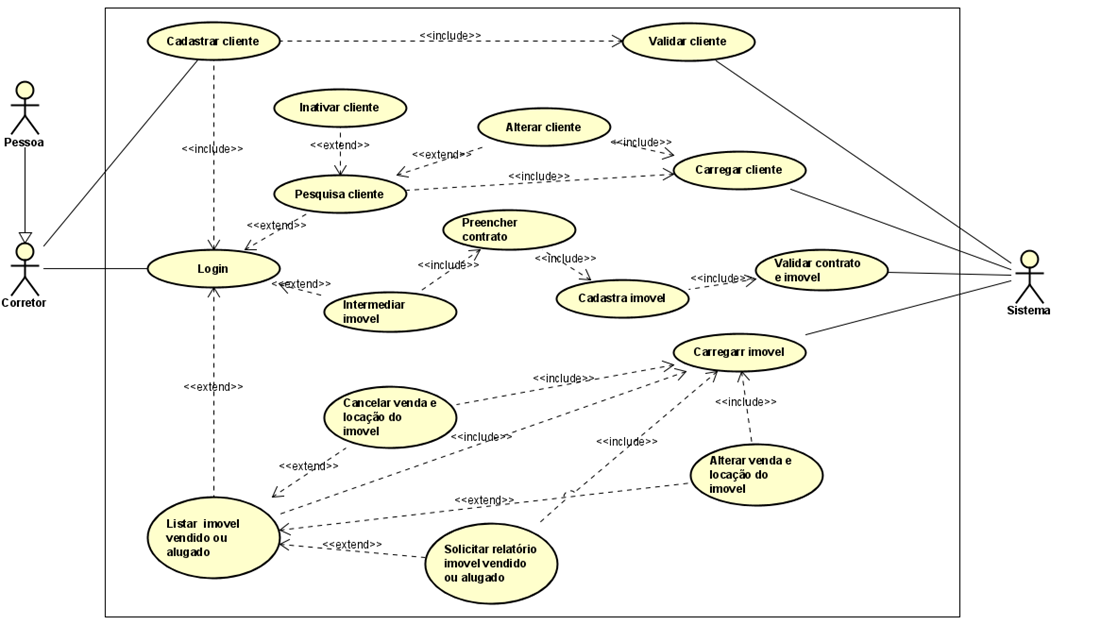
## 3.4 Lista de Casos de Uso





## 3.4. Diagrama de Casos de Uso

**Figura 01** — Diagrama de Contexto – Visão do GerenteFonte: Autoria própria.

**Figura 02** — Diagrama de Contexto – Visão do CorretorFonte: Autoria própria.

## 3.5. Diagrama de Casos de uso individuais

**Cadastro cliente:**Diagrama

Descrição gerada automaticamenteFonte: Autoria própria.

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **CU01 - Cadastrar Cliente** |
| Ator Principal | Gerente |
| Atores Secundários | Corretor |
| Resumo | Descreve etapas do gerente e corretor para cadastrar um cliente. |
| Pré-condições | Dados do cliente; ator estar logado; |
| Pós-condições |  |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1.O funcionário informa os  dados do cliente. |  |
|  | 2.Sistema valida dados do cliente. |
|  |  |
|  |  |
|  |  |
| Restrições/Validações | 1.Ser maior de idade.  2. Para realizar o cadastro é necessário ser uma pessoa física ou jurídica que é comprador, vendedor e dono de imóvel. |
| **Cenário Alternativo – “dados Inválidos”** | |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1.Sistema verifica que os dados são inválidos.  2.Sistema recusa o pedido. |
| **Cenário de Exceção – “Cliente menor de idade”** | |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1.Comunicar que a idade é invalida  2.Recusar cadastro. |
|  |  |

**Pesquisa cliente:**

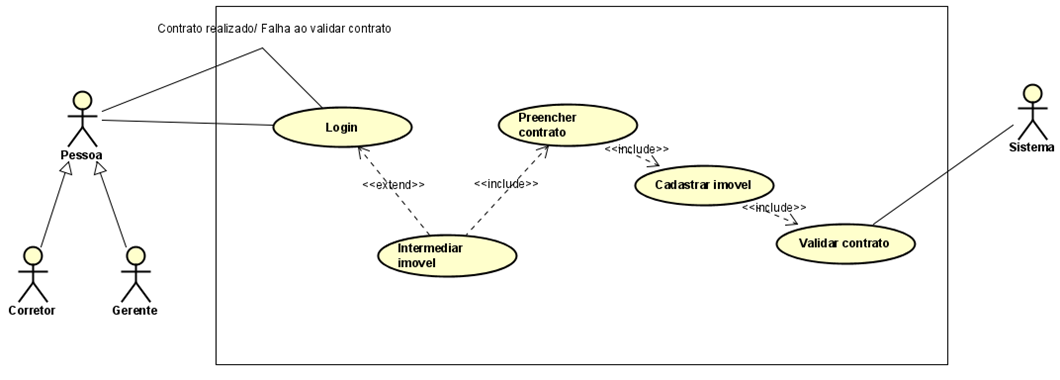
Diagrama

Descrição gerada automaticamente

Fonte: Autoria própria.

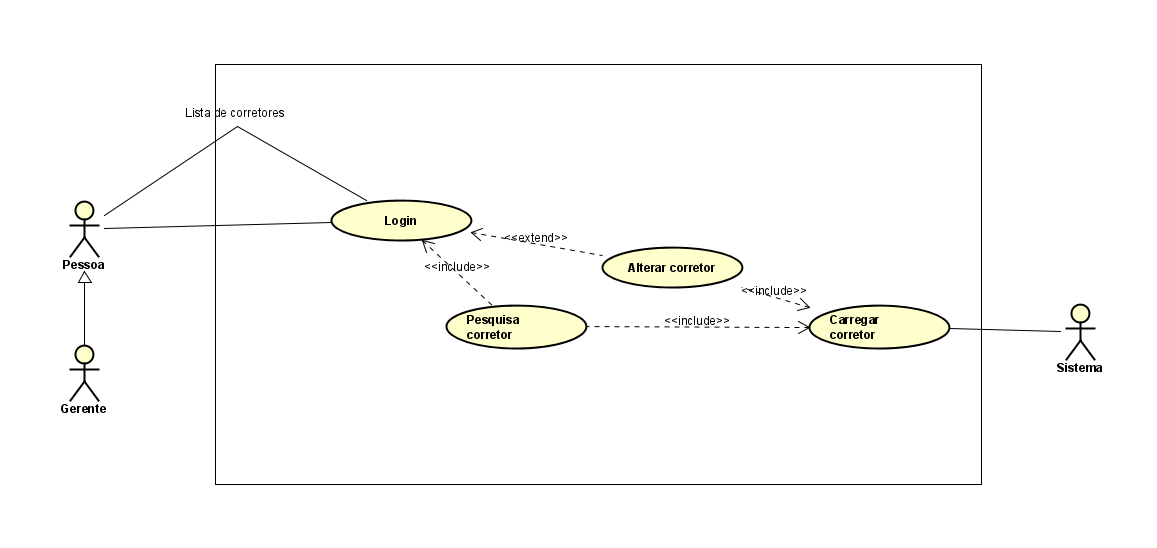
|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **CU02– Listar Cliente** |
| Ator Principal | Gerente |
| Atores Secundários | Corretor |
| Resumo | Descreve etapas do corretor e gerente para listar os clientes. |
| Pré-condições | Ator estar logado. |
| Pós-condições |  |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1.O funcionário busca os  clientes, ou informa o id de um cliente. |  |
|  | 2. Sistema carrega clientes. |
|  | 3.Sistema retorna a pesquisa. |
| Restrições/Validações | Deve ter clientes cadastrados. |
| **Cenário Alternativo – “Cenário”** | |
| **Ações do Ator** | **Ações do Sistema** |
|  |  |
| **Cenário de Exceção – “Exceção”** | |
| **Ações do Ator** | **Ações do Sistema** |
|  |  |
|  |  |

**Intermedia Imóvel:**



|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **CU03 – Intermediar Imóvel** |
| Ator Principal | Gerente |
| Atores Secundários | Corretor |
| Resumo | Descreve etapas do gerente e corretor para intermediar um imóvel. |
| Pré-condições | Ator estar logado, dados do imóvel, dados do contrato. |
| Pós-condições |  |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1.O funcionário intermedia o imóvel. |  |
| 2.O cliente preenche o contrato para compra, venda e locação de imóvel com o funcionário. |  |
| 3.Funcionario cadastra o imóvel. |  |
|  | 4. Sistema cadastra imóvel e contrato. |
| Restrições/Validações | 1.O cliente deve estar cadastrado. |
| **Cenário Alternativo – Dados inválidos** | |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. O sistema avisa que os dados do imóvel são inválidos.  2.Recusar pedido. |
| **Cenário de Exceção – Cliente não cadastrado** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Sistema avisa que o cliente não está cadastrado. | 2.Recusar cadastro. |

**Pesquisa corretor:**



|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **UC4 – Listar Funcionário** |
| Ator Principal | Gerente |
| Atores Secundários |  |
| Resumo | Descreve etapas do gerente para listar os funcionários. |
| Pré-condições | Ator estar logado. |
| Pós-condições |  |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1.O gerente busca os  funcionários, ou informa o  id de um funcionário. |  |
|  | 2.Sistema carrega funcionários. |
|  | 3 Sistema retorna a pesquisa |
|  |  |
|  |  |
| Restrições/Validações | Deve ter funcionários cadastrados. |
| **Cenário Alternativo – “Cenário”** | |
| **Ações do Ator** | **Ações do Sistema** |
|  |  |
| **Cenário de Exceção – “Exceção”** | |
| **Ações do Ator** | **Ações do Sistema** |
|  |  |

3.5.1 – Caso de uso: Cadastrar Atendimento

**Figura 02** — Diagrama de Contexto – Visão do Paciente

**

Fonte: Silva, 2016.

**Fluxo Normal**

1-AtorAdm solicita exclusão do registro da pessoa

2-Sistema envia mensagem 06: "Deseja excluir o registro?"

3-AtorAdm confirma exclusão do registro

4-Sistema exclui registro

5-Sistema envia msg07: "Registro excluído com sucesso!"

6-Sistema atualiza banco de dados e retorna ao Caso de Uso "Alterar Pessoa"

**Fluxo Exceção**

2-Sistema envia mensagem 06: "Deseja excluir o registro?"

2.1-AtorAdm não confirma exclusão do registro

2.2-Sistema cancela ação

2.3-Sistema retorna ao caso de Uso "Alterar Pessoa"

Fluxo Alternativo

4-Sistema exclui registro

4.1-Sistema identifica erros de conexão com banco de dados ao excluir registro

4.2-Sistema envia msg07:"Erros de conexão de BD"

4.3-Sistema retorna ao item 1

## 3.6. Diagrama de Sequência

*(2 principais: cadastro, consulta, relatório, agendamento, controle, etc..)*

*Inserir o diagrama e abaixo colocar a documentação do mesmo.*

## 3.7. Diagrama de Comunicação

*(1 diagrama relevante do projeto)*

## 3.8. Diagrama de Atividade

*(1 diagrama de uma ação específica do sistema)*

# DEFINIÇÃO DA INTERFACE COM O USUÁRIO (UX) (3º semstre)

## Descrição de cenário

(citar referências sobre o assunto e descrever pelo menos 2 cenários de uso relevantes para o sistema)

## Descrição de personas

(citar referências sobre o assunto e descrever pelo menos 2 personas relevantes para o sistema)

## Esboços de tela (wireframes)

(fazer o wireframe da tela de login, tela principal e dois exemplos de cadastros relevantes para o sistema e descrever sobre a imagem que foi colocada)

## Protótipes de tela

(fazer o protótipo da tela de login, tela principal e dos dois exemplos de cadastros relevantes para o sistema, preferencialmente em Html e css ou em bootstrap, se não for possível fazer o protótipo em algum aplicativo de edição pertinente)

(descrever sobre as telas demonstradas)

# BRANCO DE DADOS

## Modelo Entidade Relacionamento

*(descrever sobre o que se refere o Mapeamento do Objeto Relacional)*

Figura 9 – Mapeamento do Objeto Relacional



## Script das tabelas

# ARQUITETUdA DE SOFTWARE

## 6.1 Arquitetura de desenvolvimento

*Explanar sobre como foi realizado o processo de desenvolvimento, arquitetura de desenvolvimento (Sugestão: fazer um diagrama de Componentes para demonstrar a estrutura do sistema e suas camadas)*

## 6.2 Telas do sistema

*Colocar as telas desenvolvidas (figuras) e explicar sua função e funcionamento*

# CONCLUSÃO

Fazer uma conclusão se é viável o desenvolvimento do software.

# REFERÊNCIAS

*Colocar as referências utilizadas no projeto conforme os modelos abaixo:*

ASFHAL, C. R. **Gestão de segurança do trabalho e de saúde ocupacional**. São Paulo: Reichmann & Autores, 2005.

BEAIRD, J. Princípios do Web Design Maravilhoso. Rio de Janeiro: Altabooks, 2008.

CHIAVENATO, I. **Gestão de pessoas**: o novo papel dos recursos humanos nas organizações. 3. ed. Rio de Janeiro: Elsevier, 2008.

COSTA, H. J. Acidentes do trabalho: teremos nova lei acidentária?. **Jus Navigandi**, Teresina, v. 9, n. 664, 1 maio 2005. Disponível em: <http://jus2.uol.com.br/doutrina/texto.asp?id=6662>. Acesso em: 1 jul. 2010.

DELIBERATO, P. C. P. **Fisioterapia preventiva**: fundamentos e aplicações. São Paulo: Manole, 2002.

DIAS, E. C.; MELO, E. M. de. Políticas públicas em saúde e segurança no trabalho. In: MENDES, R. (Org.). **Minha do trabalho**. 2. ed. atual. e ampl. São Paulo: Atheneu, 2007. v. 2. p. 1683-1720.

GUEDES, Gilleanes T. A. **UML Uma Abordagem prática**, 3 ed. São Paulo: Novatec, 2008.

GUEDES, Gilleanes T. A. **UML 2 Uma Abordagem prática**, São Paulo: Novatec, 2009.

WATRALL, E; SIARTO, J. Use A Cabeça! **Web Design.** Alta Books, 2009.