

# Documento de Requisitos

## Viga mestre

[11/12/2025]

Versão 1.0.0

Juan Silva Garcia

João Victor Superbi

## Histórico de Alterações

Data	Versão	Descrição	Autor
19/09/2025	1.0.0	Draft inicial do documento.	Juan, João Victor

# Conteúdo

<b>INTRODUÇÃO.....</b>	<b>6</b>
VISÃO GERAL DO DOCUMENTO.....	6
CONVENÇÕES, TERMOS E ABREVIAÇÕES.....	6
IDENTIFICAÇÃO DOS REQUISITOS.....	7
PRIORIDADES DOS REQUISITOS.....	7
DESCRIPÇÃO GERAL DO SISTEMA.....	7
CLIENTE.....	7
USUÁRIO.....	7
VISÃO GERAL DO SISTEMA.....	8
REQUISITOS FUNCIONAIS.....	8
[RF001] CADASTRAR FUNCIONÁRIO.....	8
[RF 002] CONSULTAR FUNCIONÁRIOS.....	8
[RF 003] EDITAR FUNCIONÁRIO.....	9
[RF 004] EXCLUIR FUNCIONÁRIO.....	9
[RF 005] GERENCIAR OBRA (CRUD).....	9
[RF 006] DEFINIR CRONOGRAMA DA OBRA.....	9
[RF 007] ATRIBUIR TAREFA.....	9
[RF 008] ATUALIZAR STATUS DA TAREFA.....	9
[RF 009] GERAR RELATÓRIO DE OBRA.....	9
[RF 010] GERENCIAR SERVIÇOS CONTRATADOS (CRUD).....	10
[RF 011] GERENCIAR MATERIAIS (CRUD).....	10
[RF 012] GERENCIAR CATEGORIAS DE MATERIAIS (CRUD).....	10
[RF 013] GERENCIAR FORNECEDORES (CRUD).....	10
[RF 014] LANÇAR DESPESAS.....	10
[RF 015] GERAR RELATÓRIO DE GASTOS.....	10
[RF 016] GERAR RELATÓRIO DE OBRA.....	10
[RF 017] REGISTRAR VENDA DE APARTAMENTO.....	10
REQUISITOS NÃO FUNCIONAIS.....	11
USABILIDADE.....	11
[NF001] INTERFACE AMIGÁVEL.....	11
[NF002] COMPONENTES WEB.....	11
SOFTWARE.....	11
[NF003] BANCO DE DADOS POSTGRES.....	11
[NF004] LINGUAGEM JAVA.....	12
[NF005] REPOSITÓRIO E VERSIONAMENTO.....	12
DESEMPENHO.....	12
[NF006] AGILIDADE NA EXECUÇÃO DAS OPERAÇÕES.....	12

<i>[NF007] TEMPO DE GERAÇÃO DE RELATÓRIOS.....</i>	12
<i>SEGURANÇA.....</i>	13
<i>[NF008] AUTENTICAÇÃO DE USUÁRIOS.....</i>	13
<i>[NF009] CONTROLE DE ACESSO BASEADO EM PERFIL.....</i>	13
<i>INFRAESTRUTURA / OPERACIONAIS.....</i>	13
<i>[NF010] REQUISITOS DE SERVIDOR DE APLICAÇÃO.....</i>	13
<i>[NF011] REQUISITOS DE SERVIDOR DE BANCO DE DADOS.....</i>	13
<i>[NF012] PROTOCOLO DE ACESSO.....</i>	14
<i>DIAGRAMAS DE CASOS DE USO.....</i>	14
<i>DETALHAMENTO DOS CASOS DE USO.....</i>	15
<i>[CDU001] CADASTRAR FUNCIONÁRIO.....</i>	15
<i>[CDU002] CADASTRAR OBRA.....</i>	16
<i>[CDU003] LANÇAR DESPESAS.....</i>	17
<i>[CDU004] ATRIBUIR TAREFA.....</i>	18
<i>DIAGRAMAS DE CLASSES.....</i>	19
<i>DIAGRAMAS DE SEQUÊNCIA.....</i>	19
CADASTRO DE NOVA OBRA.....	20
ATRIBUIÇÃO DE TAREFA A FUNCIONÁRIO.....	20
VENDA DE APARTAMENTO.....	20
SOLICITAÇÃO DE MATERIAL.....	21
ATUALIZAÇÃO DE CRONOGRAMA.....	21
GERAÇÃO DE RELATÓRIO DE GASTOS.....	22
<i>DIAGRAMAS DE ATIVIDADES.....</i>	22
SOLICITAR APROVAÇÃO.....	23
GERENCIAR MATERIAIS.....	23
RELATÓRIO DE GASTO.....	24
PROCESSO DE VENDA DE APARTAMENTOS.....	24
<i>DIAGRAMAS DE ESTADOS.....</i>	25
TAREFA.....	25
OBRA.....	26
APARTAMENTO/UNIDADE.....	27
SERVICO.....	28
<i>DIAGRAMAS DE IMPLANTAÇÃO.....</i>	29
<i>NÍVEL DE ACESSO POR TIPO DE USUÁRIO.....</i>	30
<i>PONTUAÇÃO DO PROJETO.....</i>	31
CÁLCULO DO PESO DOS ATORES.....	31
CÁLCULO DO PESO DOS CASOS DE USO.....	32
PONTOS DE CASO DE USO NÃO AJUSTADOS.....	32
FATOR DE COMPLEXIDADE TÉCNICA.....	33
FATOR DE COMPLEXIDADE AMBIENTAL.....	34

<i>CÁLCULO FINAL E ESTIMATIVA DE ESFORÇO</i> .....	34
<i>DIAGRAMA DE GANTT</i> .....	34
<i>DEFINIÇÃO DO BACKLOG E PRIMEIRO SPRINT NO SCRUM</i> .....	36
<i>PRODUCT BACKLOG</i> .....	36
<i>SPRINT</i> .....	39
<i>SPRINT BACKLOG</i> .....	41

## INTRODUÇÃO

Este documento especifica os requisitos do Viga mestre, fornecendo as informações necessárias para o projeto e implementação, assim como para a realização dos testes e homologação do sistema. O objetivo deste projeto é desenvolver uma solução para a falta de comunicação e controle centralizado em projetos de construção civil, que frequentemente resulta em atrasos, desperdício de materiais e estouro de orçamento.

## VISÃO GERAL DO DOCUMENTO

Além desta seção introdutória, as seções seguintes estão organizadas como descrito abaixo.

- **Seção 2 – Descrição Geral do Sistema:** Apresenta uma visão geral do sistema, caracterizando seu escopo e descrevendo seus usuários.
- **Seção 3 – Requisitos Funcionais:** Especifica todos os cenários e funcionalidades que o sistema deve executar.
- **Seção 4 – Requisitos Não Funcionais:** Detalha os requisitos de usabilidade, software, desempenho e segurança do sistema.
- **Seção 5 – Diagramas de Casos de Uso:** Especifica os atores e os cenários do sistema utilizando a notação de diagramas UML.
- **Seção 6 – Detalhamento dos Casos de Uso:** Descreve a prioridade, o fluxo principal e os fluxos alternativos dos casos de uso e suas relações com os requisitos.
- **Seção 7 – Diagramas de Classes:** Apresenta o diagrama de classes, detalhando a estrutura estática do sistema e os relacionamentos entre as entidades.
- **Seção 8 – Diagramas de Atividades:** Ilustra os fluxos de trabalho (workflows) dos principais processos do sistema, como a solicitação de aprovação e o gerenciamento de materiais.
- **Seção 9 – Nível de Acesso por Tipo de Usuário:** Define as permissões de acesso específicas para cada perfil de usuário dentro do sistema.
- **Seção 10 – Pontuação do Projeto:** Detalha a metodologia de estimativa de esforço, incluindo o cálculo de peso dos atores, casos de uso e os fatores de complexidade técnica e ambiental.
- **Seção 11 – Diagrama de Gantt:** Apresenta o cronograma visual do projeto, com as principais fases, suas durações e prazos.
- **Seção 12 – Definição do Backlog e Primeiro Sprint no Scrum:** Organiza as funcionalidades no Product Backlog e detalha o planejamento e o escopo do primeiro Sprint de desenvolvimento.

## CONVENÇÕES, TERMOS E ABREVIACÕES

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

## IDENTIFICAÇÃO DOS REQUISITOS

Por convenção, a referência a requisitos é feita através do identificador do requisito, de acordo com a especificação a seguir:

[*identificador do requisito*]

Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RF001] para os requisitos funcionais e [NF001] para os não funcionais e prossegue sendo incrementada à medida que forem surgindo novos requisitos.

## PRIORIDADES DOS REQUISITOS

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações “essencial”, “importante” e “desejável”.

- **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.
- **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.
- **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

## DESCRIÇÃO GERAL DO SISTEMA

Esta seção descreve superficialmente o cliente, os futuros usuários e fornece uma visão geral do *VIGA MESTRE*.

### CLIENTE

O Cliente é o consumidor final e potencial comprador das unidades imobiliárias (apartamentos) geridas pela construtora. Este perfil abrange pessoas físicas ou jurídicas interessadas na aquisição de imóveis, que buscam transparência nas negociações e autonomia para simular custos. No contexto do sistema Viga Mestre, o Cliente é um usuário externo que acessa a plataforma para visualizar unidades, realizar simulações financeiras (cálculo de impostos) e formalizar a compra.

### USUÁRIO

Os usuários do sistema Viga Mestre são os colaboradores da construtora e são categorizados nos seguintes perfis, cada um com permissões específicas para suas funções:

- **Administrador:** Possui acesso total ao sistema. É responsável pelo gerenciamento de cadastros essenciais, como funcionários, fornecedores e materiais, além de configurar os parâmetros gerais da plataforma.

- **Gerente de Obras:** Responsável pela visão macro do projeto. Utiliza o sistema para cadastrar novas obras, definir o cronograma principal, contratar serviços e extrair relatórios gerenciais para tomada de decisão.
- **Engenheiro:** Atua no nível tático da obra. Sua principal função no sistema é detalhar o cronograma, atribuindo tarefas específicas aos funcionários e acompanhando o progresso técnico.
- **Mestre de Obras:** Supervisiona a execução no canteiro. Usa o sistema para solicitar materiais para o estoque e para atualizar o andamento das tarefas da equipe.
- **Funcionário (Pedreiro, Pintor, etc.):** Perfil operacional. Utiliza a plataforma para visualizar as tarefas que lhe foram atribuídas e para atualizar o status delas (ex: "Em Andamento", "Concluída").
- **Financeiro:** Responsável pelo controle de custos. Utiliza o sistema para lançar todas as despesas relacionadas a uma obra e para gerar relatórios de gastos detalhados.

## VISÃO GERAL DO SISTEMA

O VIGA MESTRE tem, como principal objetivo, oferecer uma plataforma integrada para o gerenciamento de obras, centralizando as ações de diferentes perfis de usuário. A ideia é automatizar e simplificar processos, utilizando para isso um sistema onde cada ator, do Administrador ao Pedreiro, possui acesso às funcionalidades pertinentes à sua função.

Para realizar o gerenciamento da obra, o Gerente de Obras pode registrar o terreno, contratar serviços e organizar o cronograma, dividindo-o em 2 macro-estágios. A partir disso, o Engenheiro pode atribuir tarefas específicas a cada Funcionário (Pedreiro, Pintor, Servente), que por sua vez, podem atualizar o status de suas atividades, garantindo que o andamento do projeto seja sempre transparente.

O sistema garante a integridade dos dados e evita conflitos. Por exemplo, não será possível atribuir o mesmo funcionário a duas tarefas no mesmo horário. Além disso, o Mestre de Obras poderá solicitar material diretamente pela plataforma, cujo controle é gerenciado pelo Administrador, que também classifica os materiais por categoria em conjunto com os fornecedores.

A plataforma permite a geração de informações cruciais para a tomada de decisão: o Administrador e o Gerente de Obras podem gerar relatórios de obra, enquanto o setor Financeiro é responsável por lançar despesas e consolidar o relatório de gastos. Finalmente, o sistema também apoia a área comercial, permitindo ao Cliente/Comprador calcular impostos e realizar a venda de apartamentos diretamente pela plataforma.

## REQUISITOS FUNCIONAIS

Módulo: Gestão de Pessoas

### [RF001] CADASTRAR FUNCIONÁRIO

**Descrição:** O sistema deve permitir que o Administrador ou Gerente de Obras cadastre novos funcionários, informando dados pessoais e sua função específica (Engenheiro, Pedreiro, Pintor, etc.).

### [RF 002] CONSULTAR FUNCIONÁRIOS

**Descrição:** O sistema deve fornecer uma interface para buscar, filtrar e visualizar todos os funcionários cadastrados e suas informações.

---

#### [RF 003] EDITAR FUNCIONÁRIO

**Descrição:** O sistema deve permitir a alteração dos dados de um funcionário já cadastrado.

---

#### [RF 004] EXCLUIR FUNCIONÁRIO

**Descrição:** O sistema deve permitir que o Administrador realize a exclusão de funcionários que não estão mais ativos na empresa.

Módulo: Gestão de Obras

---

#### [RF 005] GERENCIAR OBRA (CRUD)

**Descrição:** O sistema deve permitir que o Administrador ou Gerente de Obras crie, consulte, edite e exclua (ou arquive) os registros das obras. O cadastro deve incluir informações como endereço, cliente e registro do terreno.

---

#### [RF 006] DEFINIR CRONOGRAMA DA OBRA

**Descrição:** O sistema deve permitir que o Gerente de Obras crie e organize o cronograma de uma obra, definindo seus 2 macro-estágios principais.

---

#### [RF 007] ATRIBUIR TAREFA

**Descrição:** O sistema deve permitir que um Engenheiro ou Gerente de Obras atribua uma tarefa do cronograma a um funcionário específico.

---

#### [RF 008] ATUALIZAR STATUS DA TAREFA

**Descrição:** O sistema deve permitir que os funcionários (Pedreiro, Pintor, Mestre de Obras) atualizem o status das tarefas que lhes foram atribuídas (ex: "A Fazer", "Em Andamento", "Bloqueada", "Concluída").

---

#### [RF 009] GERAR RELATÓRIO DE OBRA

**Descrição:** O sistema deve permitir a geração de um relatório gerencial sobre o andamento geral de uma obra, incluindo avanço físico e resumo financeiro.

---

#### [RF 010] GERENCIAR SERVIÇOS CONTRATADOS (CRUD)

**Descrição:** O sistema deve permitir que o Gerente de Obras cadastre, consulte, edite e exclua serviços de terceiros contratados para a obra, associando custos e fornecedores.

Módulo: Materiais e Fornecedores

---

#### [RF 011] GERENCIAR MATERIAIS (CRUD)

**Descrição:** O sistema deve permitir o cadastro, consulta, edição e exclusão de materiais, controlando informações como nome, unidade, categoria e quantidade em estoque.

---

#### [RF 012] GERENCIAR CATEGORIAS DE MATERIAIS (CRUD)

**Descrição:** O sistema deve permitir a criação, edição e exclusão de categorias para os materiais (ex: Elétrico, Hidráulico, Acabamento).

---

#### [RF 013] GERENCIAR FORNECEDORES (CRUD)

**Descrição:** O sistema deve permitir o cadastro completo de fornecedores, incluindo nome, contato e os materiais que fornecem.

Módulo: Financeiro e Vendas

---

#### [RF 014] LANÇAR DESPESAS

**Descrição:** O sistema deve permitir que o perfil Financeiro registre todas as despesas relacionadas a uma obra, seja de materiais ou serviços contratados.

---

#### [RF 015] GERAR RELATÓRIO DE GASTOS

**Descrição:** O sistema deve ser capaz de gerar relatórios financeiros detalhados por obra, consolidando todas as despesas lançadas em um determinado período.

---

#### [RF 016] GERAR RELATÓRIO DE OBRA

**Descrição:** O sistema deve permitir a geração de um relatório gerencial sobre o andamento geral de uma obra, incluindo avanço físico e resumo financeiro.

---

#### [RF 017] REGISTRAR VENDA DE APARTAMENTO

**Descrição:** O sistema deve permitir registrar a venda de um apartamento de uma obra a um cliente, armazenando os detalhes da transação.

## REQUISITOS NÃO FUNCIONAIS

### USABILIDADE

Esta seção descreve os requisitos não funcionais associados à facilidade de uso da interface com o usuário e *help on-line*.

#### [NF001] INTERFACE AMIGÁVEL

O sistema terá uma interface amigável ao usuário primário sem se tornar cansativa aos usuários mais experientes.

**Prioridade:**

Essencial

Importante

Desejável

**Requisitos associados:** [RF001], [RF002], [RF003], [RF004], [RF005], [RF006], [RF007], [RF008], [RF009], [RF010], [RF011], [RF012], [RF013], [RF014] e [RF015].

#### [NF002] COMPONENTES WEB

A interface deverá utilizar elementos comuns a usuários de sistemas web, como campos de texto, *combo-boxes*, *links* e botões, sem muito rebuscamento. A idéia é focar nos aspectos operacionais sem se preocupar tanto com a beleza da tela, de modo a facilitar o uso por usuários iniciantes.

**Prioridade:**

Essencial

Importante

Desejável

**Requisitos associados:** [RF001], [RF002], [RF003], [RF004], [RF005], [RF006], [RF007], [RF008], [RF009], [RF010], [RF011], [RF012], [RF013], [RF014], [RF015], [RF016], [RF017].

### SOFTWARE

Esta seção descreve os requisitos não funcionais associados aos softwares que devem ser utilizados para o desenvolvimento do sistema.

#### [NF003] BANCO DE DADOS POSTGRES

O sistema deve utilizar um banco de dados Postgres para fazer o armazenamento de dados.

**Prioridade:**

Essencial

Importante

Desejável

**Requisitos associados:** [RF001], [RF002], [RF003], [RF004], [RF005], [RF006], [RF007], [RF008], [RF009], [RF010], [RF011], [RF012], [RF013], [RF014], [RF015], [RF016], [RF017].

#### [NF004] LINGUAGEM JAVA

Visando criar um produto com maior extensibilidade, reusabilidade e flexibilidade, deve-se adotar Java como linguagem principal de desenvolvimento, seguindo cuidadosamente as técnicas de orientação a objetos.

**Prioridade:**

Essencial

Importante

Desejável

**Requisitos associados:** [RF001], [RF002], [RF003], [RF004], [RF005], [RF006], [RF007], [RF008], [RF009], [RF010], [RF011], [RF012], [RF013], [RF014], [RF015], [RF016], [RF017].

#### [NF005] REPOSITÓRIO E VERSIONAMENTO

Todo o código-fonte e documentação técnica devem ser versionados utilizando Git e hospedados no repositório oficial do projeto no GitHub, acessível através da URL: <https://github.com/Juanxit/Viga-Mestre>

**Prioridade:**

Essencial

Importante

Desejável

**Requisitos associados:** Todos.

#### DESEMPENHO

Esta seção descreve os requisitos não funcionais associados à eficiência, uso de recursos e tempo de resposta do sistema.

#### [NF006] AGILIDADE NA EXECUÇÃO DAS OPERAÇÕES

O sistema deve executar as operações no menor tempo possível, visando dar uma maior agilidade ao processo.

**Prioridade:**

Essencial

Importante

Desejável

**Requisitos associados:** [RF001], [RF002], [RF003], [RF004], [RF005], [RF006], [RF007], [RF008], [RF009], [RF010], [RF011], [RF012], [RF013], [RF014], [RF015], [RF016], [RF017].

#### [NF007] TEMPO DE GERAÇÃO DE RELATÓRIOS

O sistema deve gerar relatórios financeiros e de andamento da obra em no máximo 10 segundos, mesmo com um grande volume de dados, para não comprometer a tomada de decisão.

**Prioridade:**  Essencial  Importante  Desejável

**Requisitos associados:** [RF013], [RF014].

## SEGURANÇA

### [NF008] AUTENTICAÇÃO DE USUÁRIOS

Todo acesso ao sistema deve ser precedido de uma autenticação por login e senha. A senha deve ser armazenada de forma criptografada no banco de dados.

**Prioridade:**  Essencial  Importante  Desejável

### [NF0009] CONTROLE DE ACESSO BASEADO EM PERFIL

O sistema deve garantir que os usuários só possam acessar as funcionalidades e visualizar os dados permitidos para o seu perfil (ex: Financeiro não pode atribuir tarefas, Funcionário não pode ver relatório de gastos).

**Prioridade:**  Essencial  Importante  Desejável

## INFRAESTRUTURA / OPERACIONAIS

### INFRAESTRUTURA / OPERACIONAIS

Esta seção descreve os requisitos não funcionais associados ao ambiente de hardware e rede necessários para a execução estável do sistema.

### [NF0010] REQUISITOS DE SERVIDOR DE APLICAÇÃO

O sistema deve ser implantado em um servidor dedicado, com capacidade mínima para suportar o ambiente Java (NF004) e atender aos requisitos de desempenho (NF006).

Prioridade:  Essencial  Importante  Desejável

Requisitos associados: [NF004], [NF006].

### [NF0011] REQUISITOS DE SERVIDOR DE BANCO DE DADOS

O sistema deve ter acesso a um servidor dedicado para o banco de dados Postgres (NF003), com capacidade mínima de RAM e armazenamento em disco suficiente para suportar o volume de dados e o tempo de geração de relatórios (NF007).

Prioridade: ■ Essencial □ Importante □ Desejável

Requisitos associados: [NF003], [NF007].

---

#### [NF0012] PROTOCOLO DE ACESSO

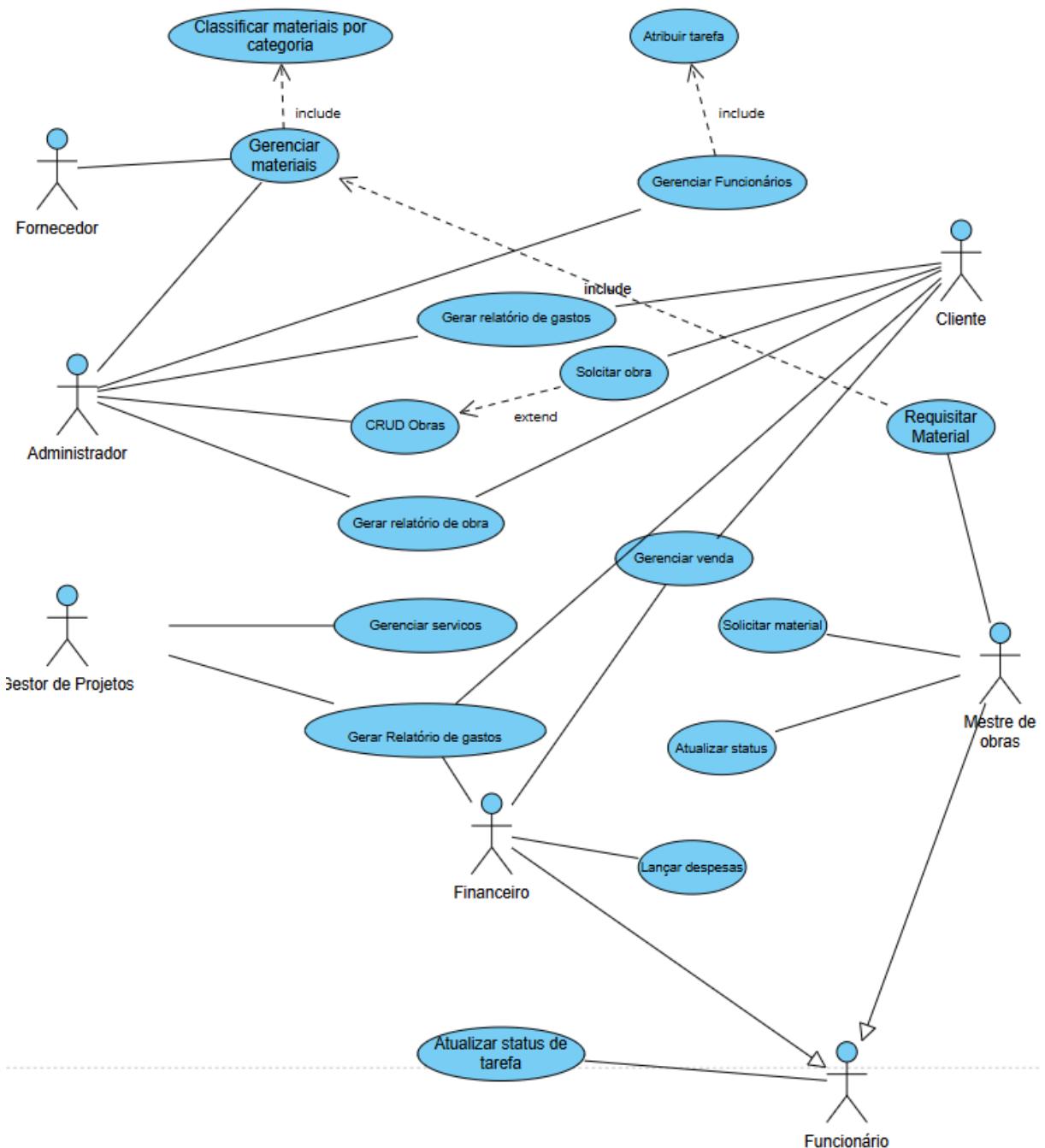
O acesso ao sistema pelos usuários (Nós Clientes) deve ser feito utilizando o protocolo HTTPS para garantir a segurança das informações transmitidas pela rede.

Prioridade: ■ Essencial □ Importante □ Desejável

Requisitos associados: [NF008], [NF009].

---

#### DIAGRAMAS DE CASOS DE USO



## DETALHAMENTO DOS CASOS DE USO

### [CDU001] CADASTRAR FUNCIONÁRIO

**Descrição do caso de uso:** O sistema deve permitir o cadastro das informações de um novo funcionário.

**Autor:** Administrador

**Prioridade:**      ■ Essencial       Importante       Desejável

**Requisitos Associados:** [RF001]

**Entradas e pré-condições:**

- O Administrador deve estar autenticado no sistema e ter permissão para gerenciar funcionários.

**Fluxo de Eventos (Fluxo Principal):**

1. O ator acessa a funcionalidade "Gerenciar Funcionários".
2. O ator seleciona a opção para adicionar um novo funcionário.
3. O sistema exibe o formulário de cadastro de funcionário (solicitando nome, CPF, cargo/função, dados de contato, etc.).
4. O ator preenche todos os campos obrigatórios.
5. O ator confirma a operação (clicando em "Salvar").
6. O sistema valida os dados inseridos.
7. O sistema armazena as informações do novo funcionário no banco de dados.
8. O sistema exibe uma mensagem de sucesso e atualiza a lista de funcionários.

**Fluxos Alternativos:**

- **[FA01] Dados inválidos:** Caso algum dado obrigatório não seja preenchido ou esteja em formato incorreto na etapa 6, o sistema exibirá uma mensagem de erro específica e não salvará o registro, permitindo que o ator corrija as informações.
- **[FA02] Funcionário já cadastrado:** Caso o CPF informado já exista no sistema, uma mensagem de erro será exibida, impedindo o cadastro em duplicidade.

**Saídas e pós-condições:**

- Um novo funcionário é cadastrado no sistema e fica disponível para ser alocado em tarefas.

---

## [CDU002] CADASTRAR OBRA

**Descrição do caso de uso:** O sistema deve permitir que o Administrador cadastre uma nova obra, que é a entidade central do sistema.

**Autor:** Administrador

**Prioridade:**      ■ Essencial       Importante       Desejável

**Requisitos Associados:** [RF005]

**Entradas e pré-condições:**

- O Administrador deve estar autenticado no sistema.
- O Cliente para o qual a obra será realizada já deve estar cadastrado no sistema.

**Fluxo de Eventos (Fluxo Principal):**

---

1. O ator acessa a funcionalidade "CRUD Obras".
2. O ator seleciona a opção para adicionar uma nova obra.
3. O sistema exibe o formulário de cadastro de obra.
4. O ator preenche as informações da obra (Nome, endereço, seleciona o Cliente associado, etc.).
5. O ator confirma a operação.
6. O sistema valida e salva os dados da nova obra.
7. O sistema exibe uma mensagem de "Obra cadastrada com sucesso".

#### **Fluxos Alternativos:**

- **[FA01] Cliente não encontrado:** Se o cliente desejado não estiver na lista, o ator deverá cancelar a operação e primeiro realizar o cadastro do cliente.

#### **Saídas e pós-condições:**

- A nova obra é criada no sistema, pronta para ter serviços, funcionários e despesas associados a ela.

---

## **[CDU003] LANÇAR DESPESAS**

**Descrição do caso de uso:** O sistema deve permitir que o setor Financeiro registre despesas (custos) e as associe a uma obra específica.

**Ator:** Financeiro

**Prioridade:**  Essencial  Importante  Desejável

**Requisitos Associados:** [RF012], [RF013]

#### **Entradas e pré-condições:**

- O ator Financeiro deve estar autenticado no sistema.
- A obra à qual a despesa se refere deve estar cadastrada.

#### **Fluxo de Eventos (Fluxo Principal):**

1. O ator acessa a funcionalidade "Lançar Despesas".
2. O sistema exibe o formulário de lançamento.
3. O ator seleciona a obra correspondente.
4. O ator preenche os detalhes da despesa (descrição, valor, data, fornecedor, etc.).
5. O ator anexa um comprovante digital (nota fiscal), se houver.
6. O ator confirma o lançamento.
7. O sistema valida e armazena a despesa, vinculando-a ao centro de custo da obra selecionada.
8. O sistema exibe uma mensagem de sucesso.

#### **Fluxos Alternativos:**

- **[FA01] Valor inválido:** Se o valor inserido não for um número válido, o sistema exibirá um erro e solicitará a correção.

#### Saídas e pós-condições:

- A despesa é registrada.
- O "Relatório de Gastos" da obra correspondente é atualizado automaticamente.

---

### [CDU004] ATRIBUIR TAREFA

**Descrição do caso de uso:** O sistema deve permitir que o Engenheiro ou Gerente de Obras atribua uma tarefa do cronograma a um ou mais funcionários.

**Autor:** Engenheiro / Gerente de Obras

**Prioridade:**  Essencial  Importante  Desejável

**Requisitos Associados:** [RF007]

#### Entradas e pré-condições:

- O ator deve estar autenticado no sistema.
- A obra e seu cronograma já devem estar cadastrados.
- A tarefa a ser atribuída já deve existir no cronograma.
- O funcionário a ser alocado deve estar cadastrado e disponível.

#### Fluxo de Eventos (Fluxo Principal):

1. O ator acessa a funcionalidade "Cronograma da Obra".
2. O ator seleciona uma tarefa específica.
3. O ator seleciona a opção "Atribuir Responsável".
4. O sistema exibe uma lista de funcionários disponíveis com a função adequada para a tarefa.
5. O ator seleciona um ou mais funcionários da lista.
6. O ator confirma a atribuição.
7. O sistema valida se o funcionário já não está alocado em outra tarefa no mesmo período.
8. O sistema vincula o funcionário à tarefa e o notifica.
9. O sistema exibe uma mensagem de sucesso.

#### Fluxos Alternativos:

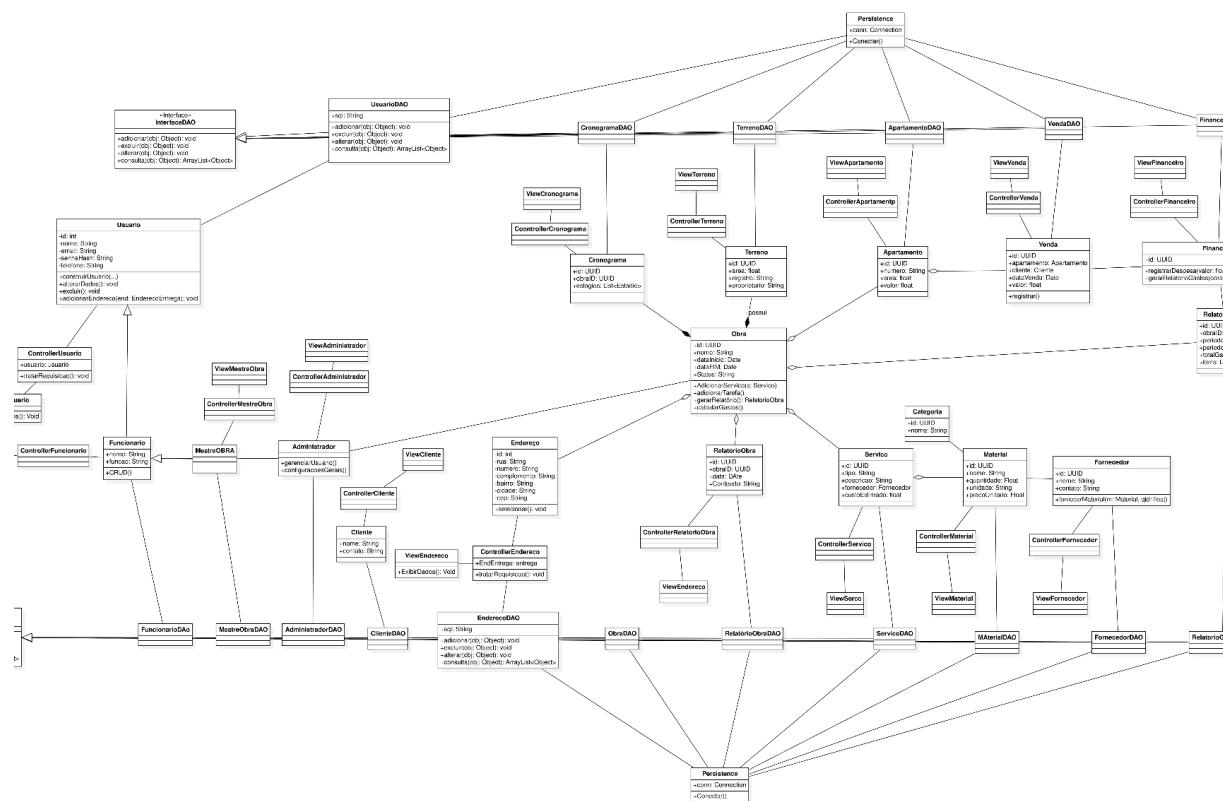
- **[FA01] Funcionário indisponível:** Caso o funcionário selecionado já esteja alocado em outra tarefa no mesmo período na etapa 7, o sistema exibirá uma mensagem de erro informando o conflito e não permitirá a atribuição.

#### Saídas e pós-condições:

- A tarefa passa a ter um ou mais responsáveis por sua execução.
- O funcionário atribuído pode visualizar a tarefa em seu painel.

## DIAGRAMAS DE CLASSES

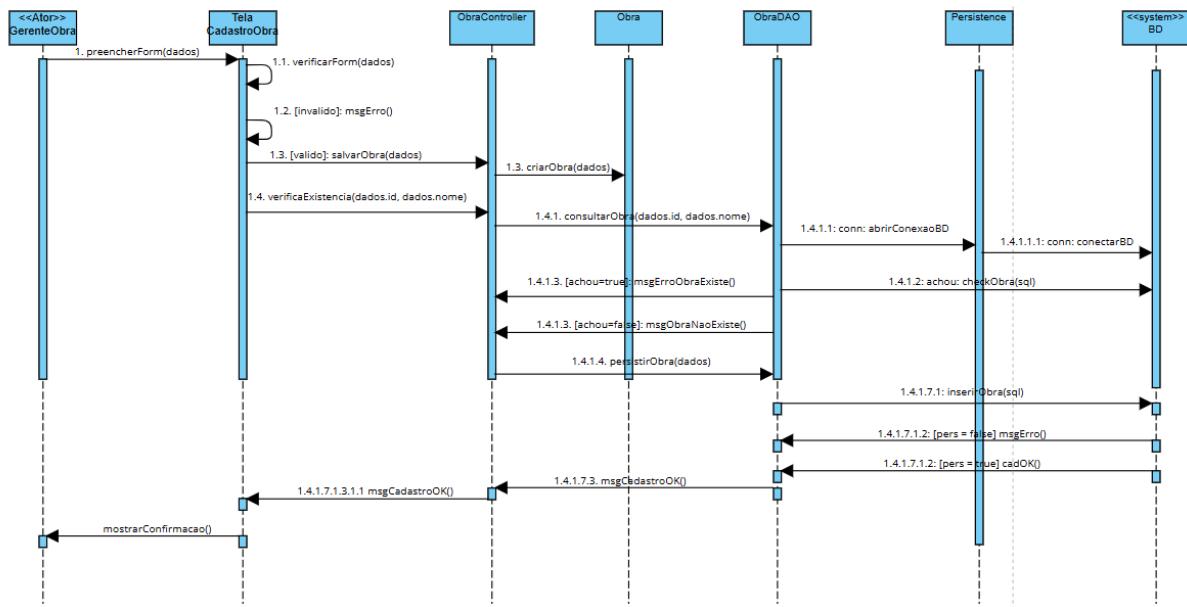
Figura 5: Diagrama de Classes



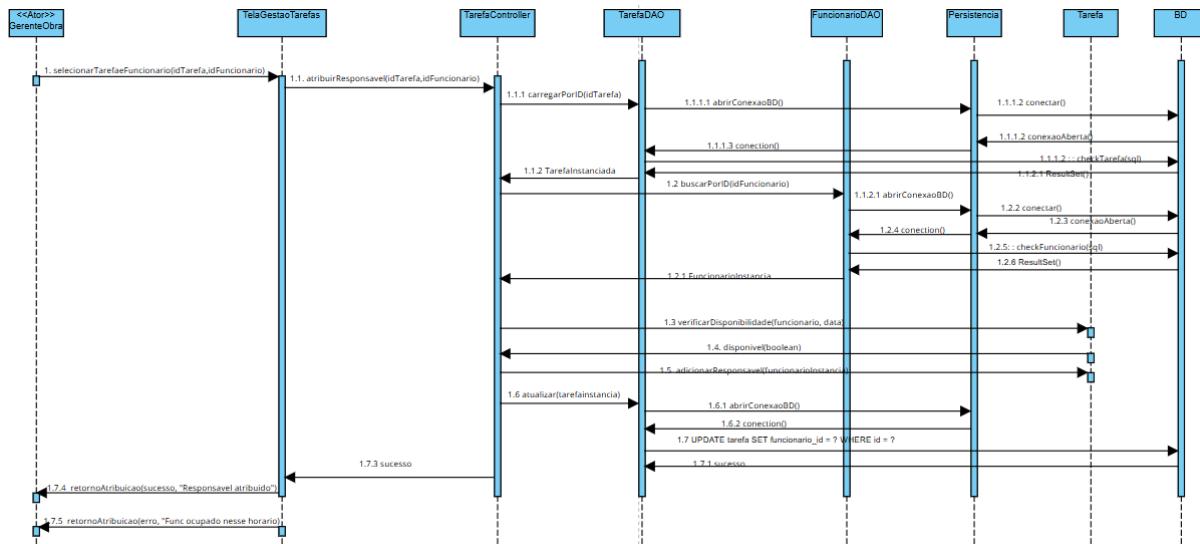
Fonte: autor

## DIAGRAMAS DE SEQUÊNCIA

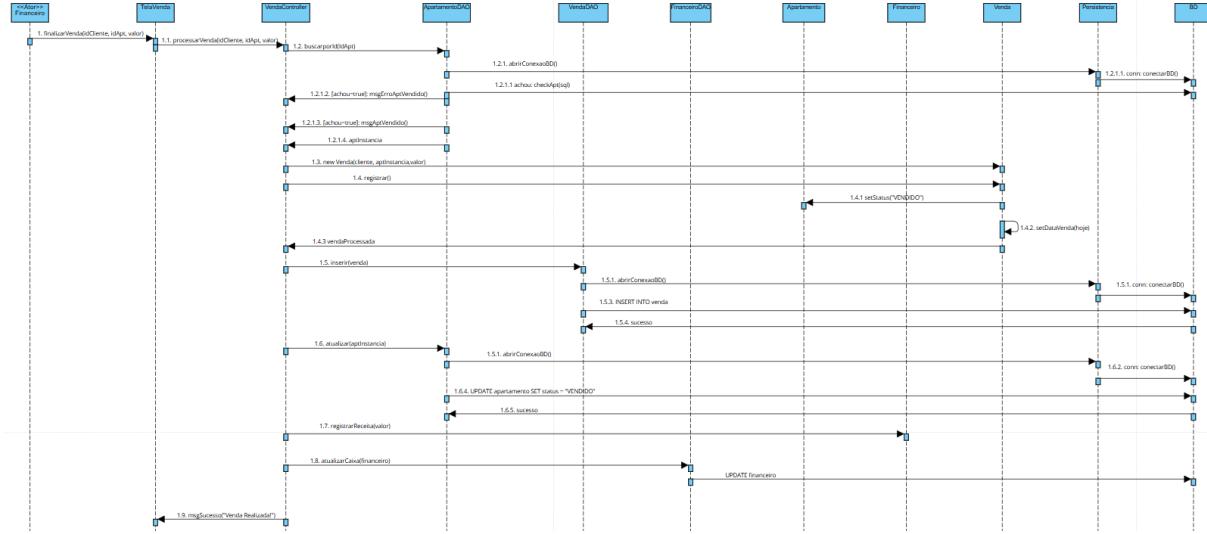
## CADASTRO DE NOVA OBRA



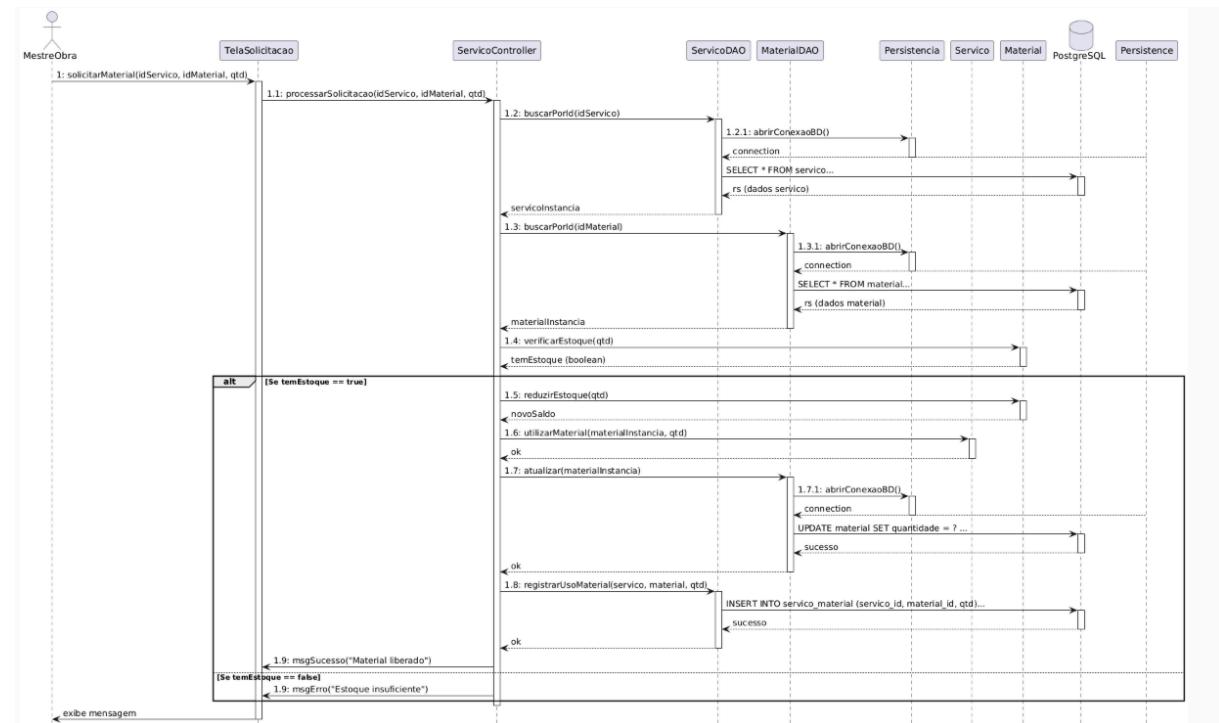
## ATRIBUIÇÃO DE TAREFA A FUNCIONÁRIO



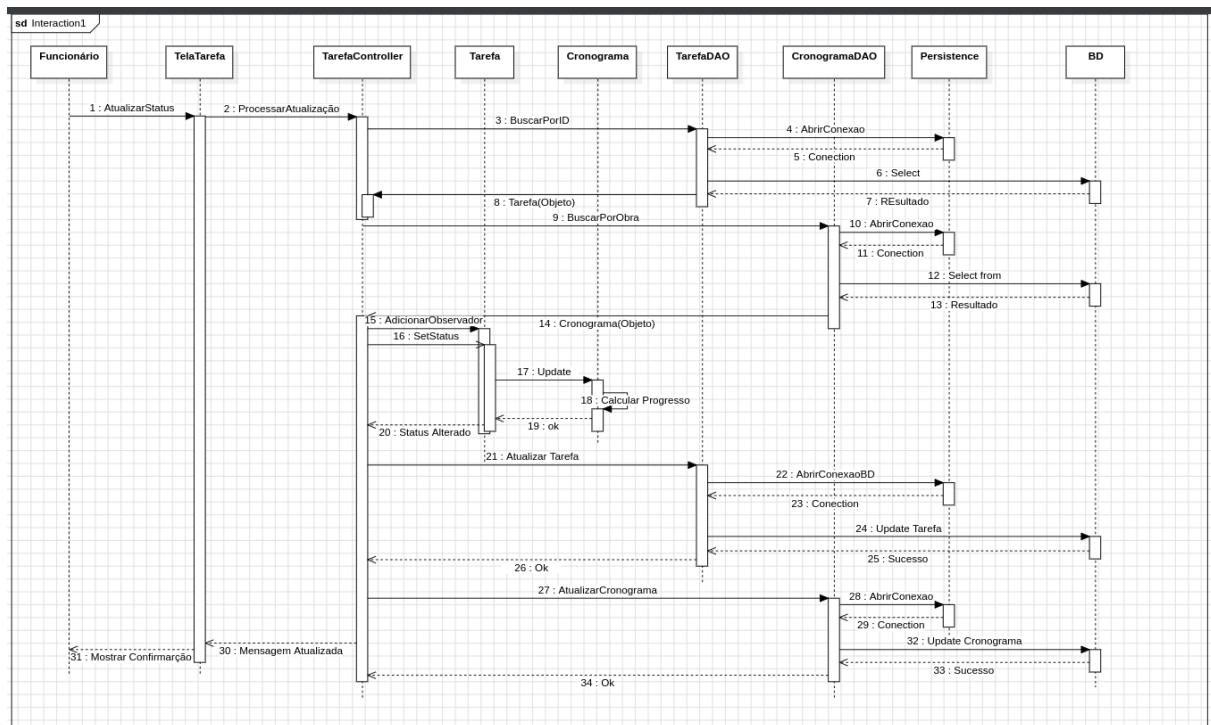
## VENDA DE APARTAMENTO



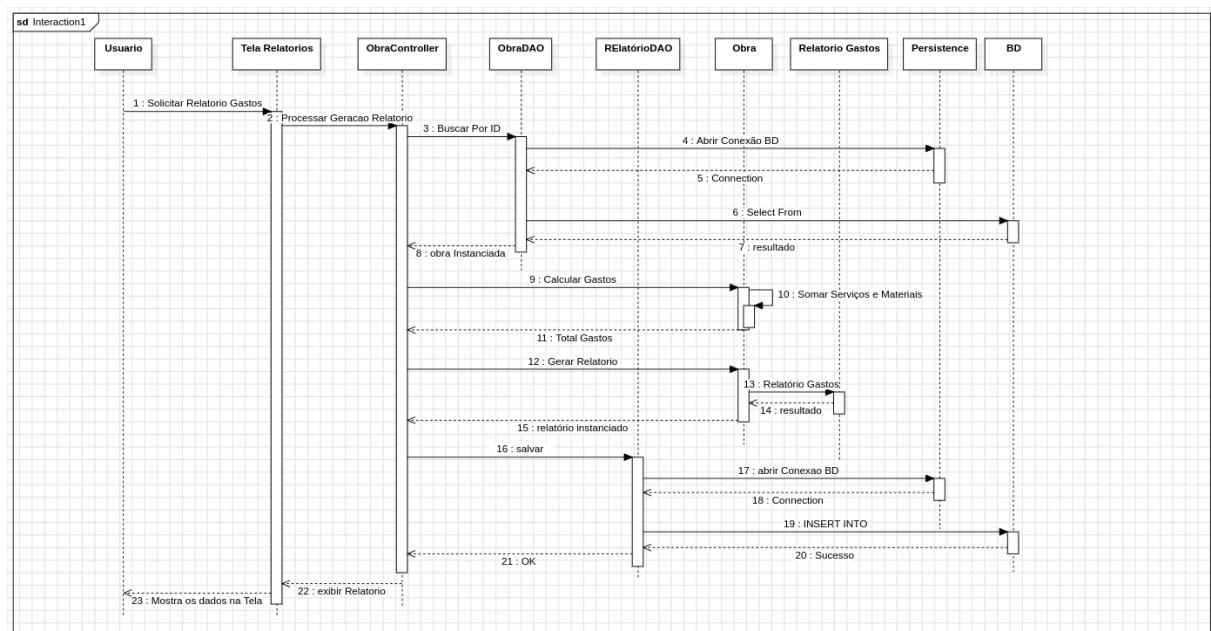
## SOLICITAÇÃO DE MATERIAL



## ATUALIZAÇÃO DE CRONOGRAMA



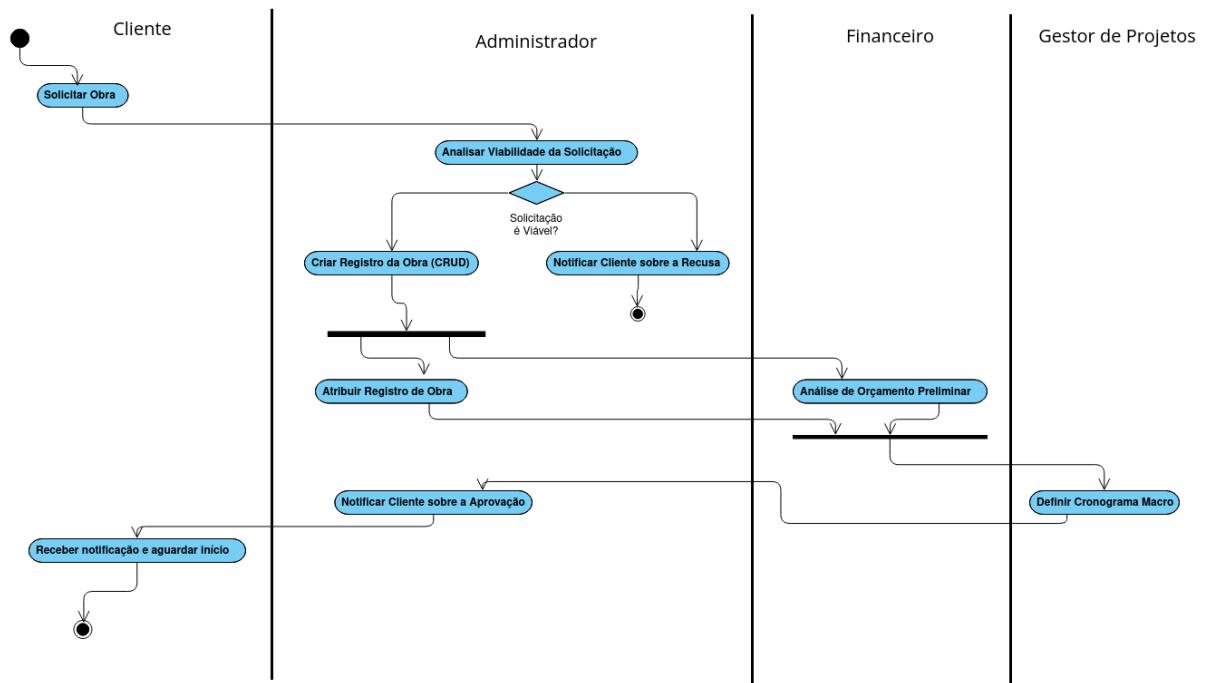
## GERAÇÃO DE RELATÓRIO DE GASTOS



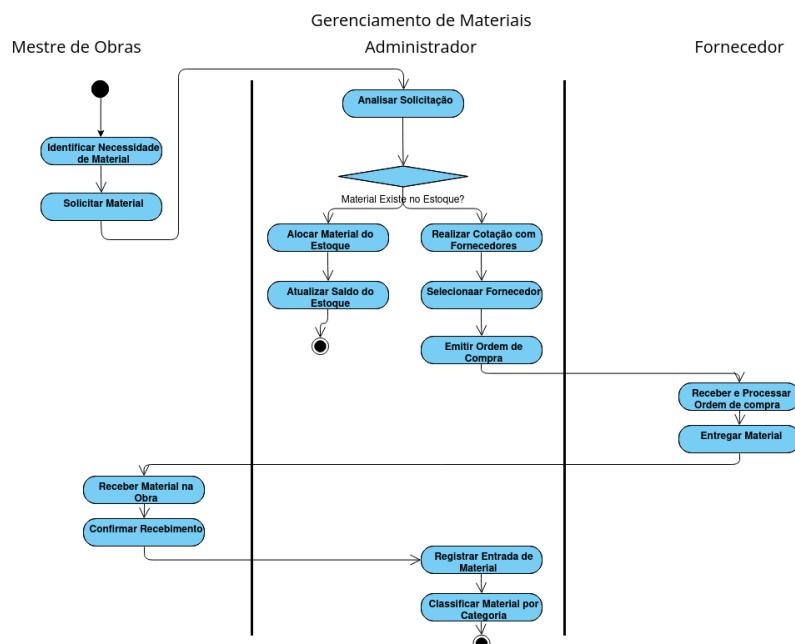
## DIAGRAMAS DE ATIVIDADES

## SOLICITAR APROVAÇÃO

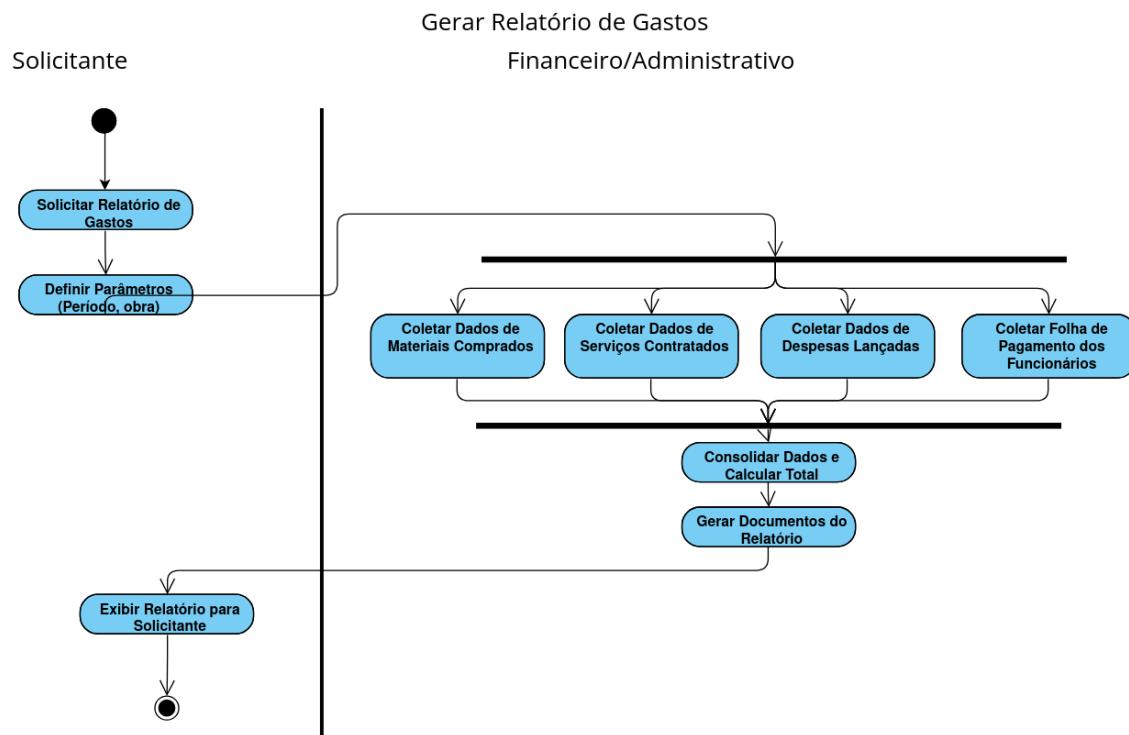
Figura 6: Solicitar aprovação



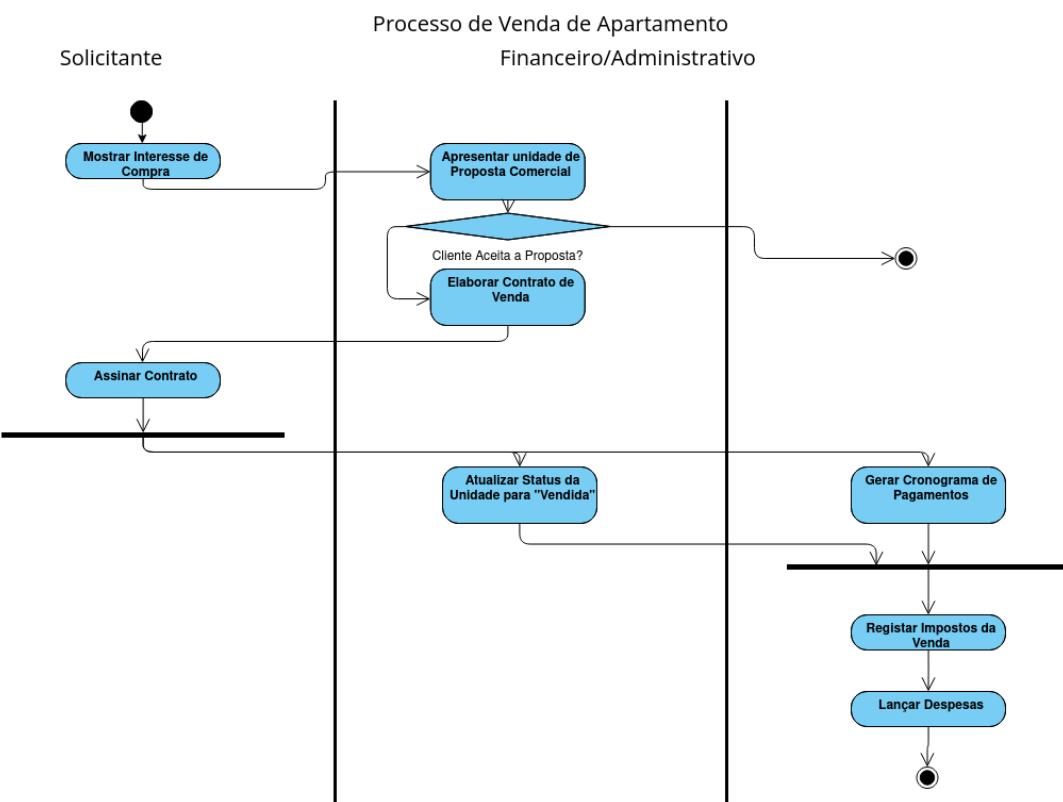
## GERENCIAR MATERIAIS



## RELATÓRIO DE GASTO

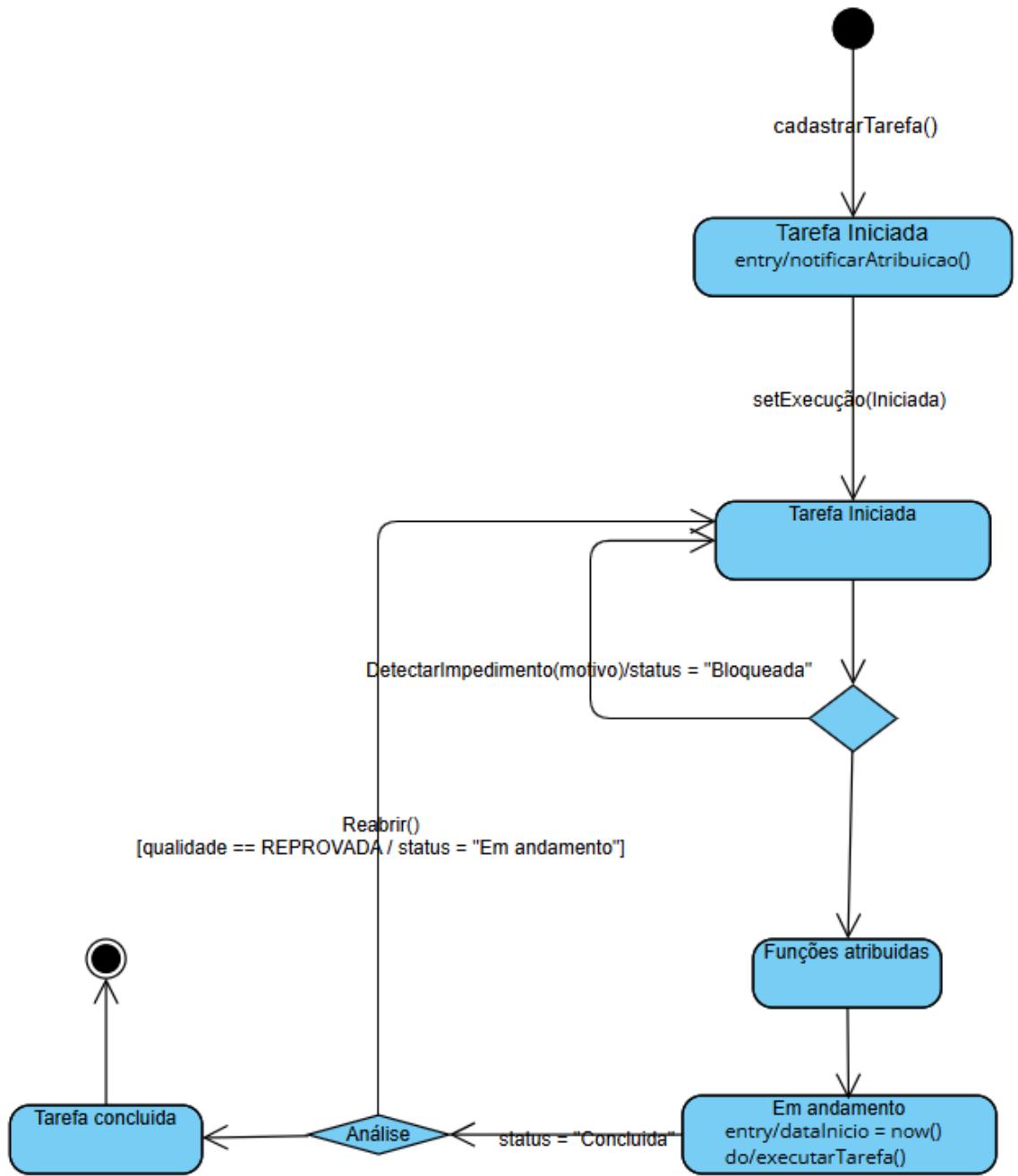


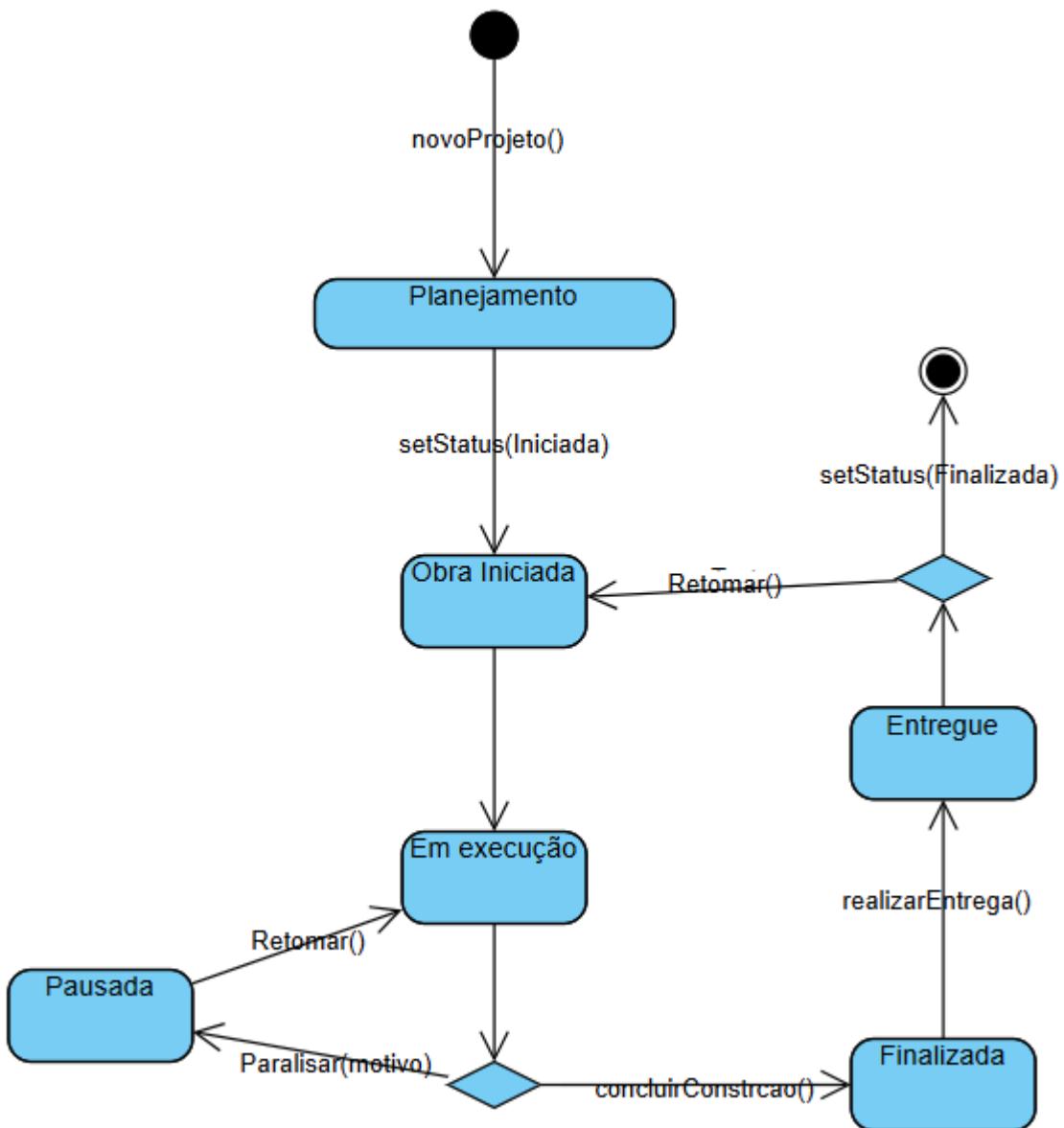
## PROCESSO DE VENDA DE APARTAMENTOS



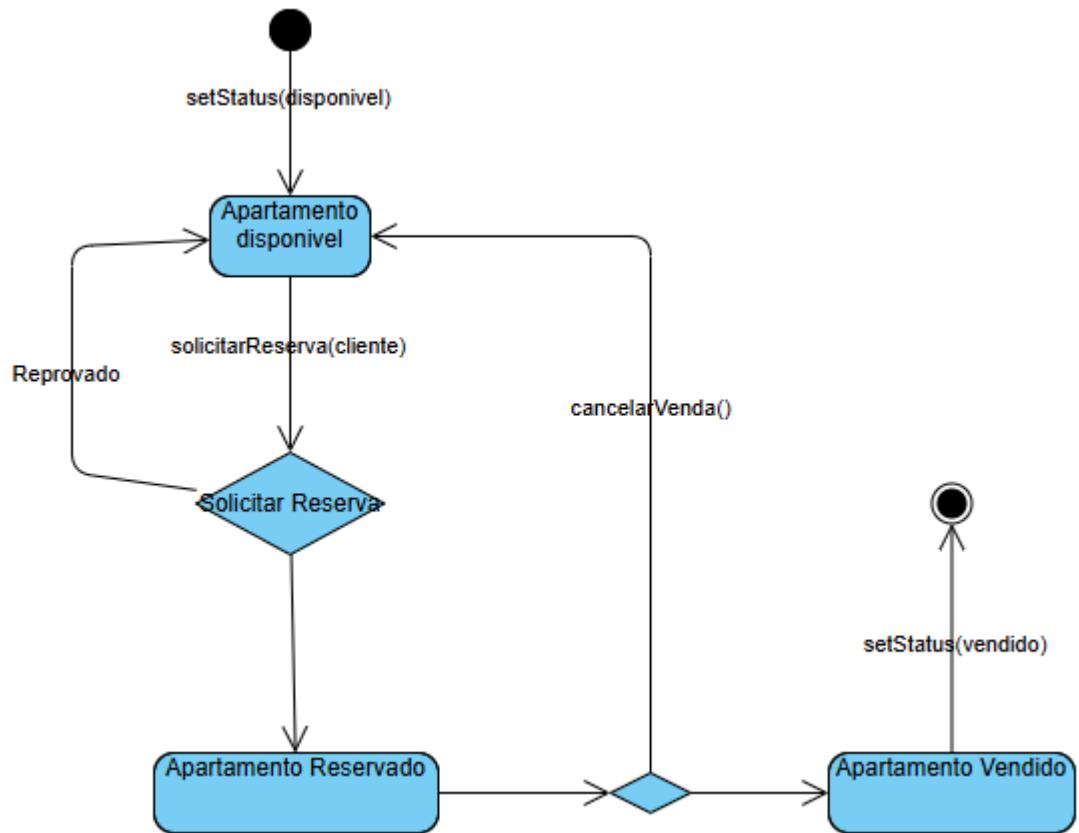
## DIAGRAMAS DE ESTADOS

### TAREFA

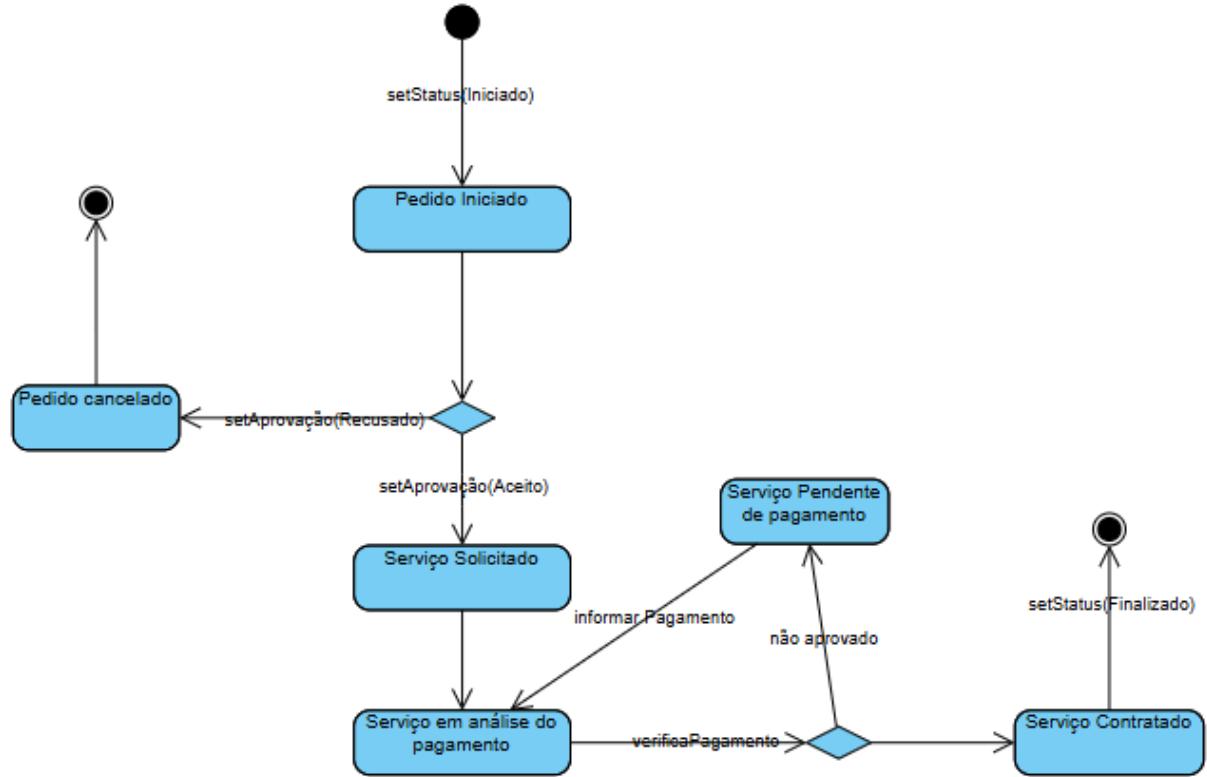




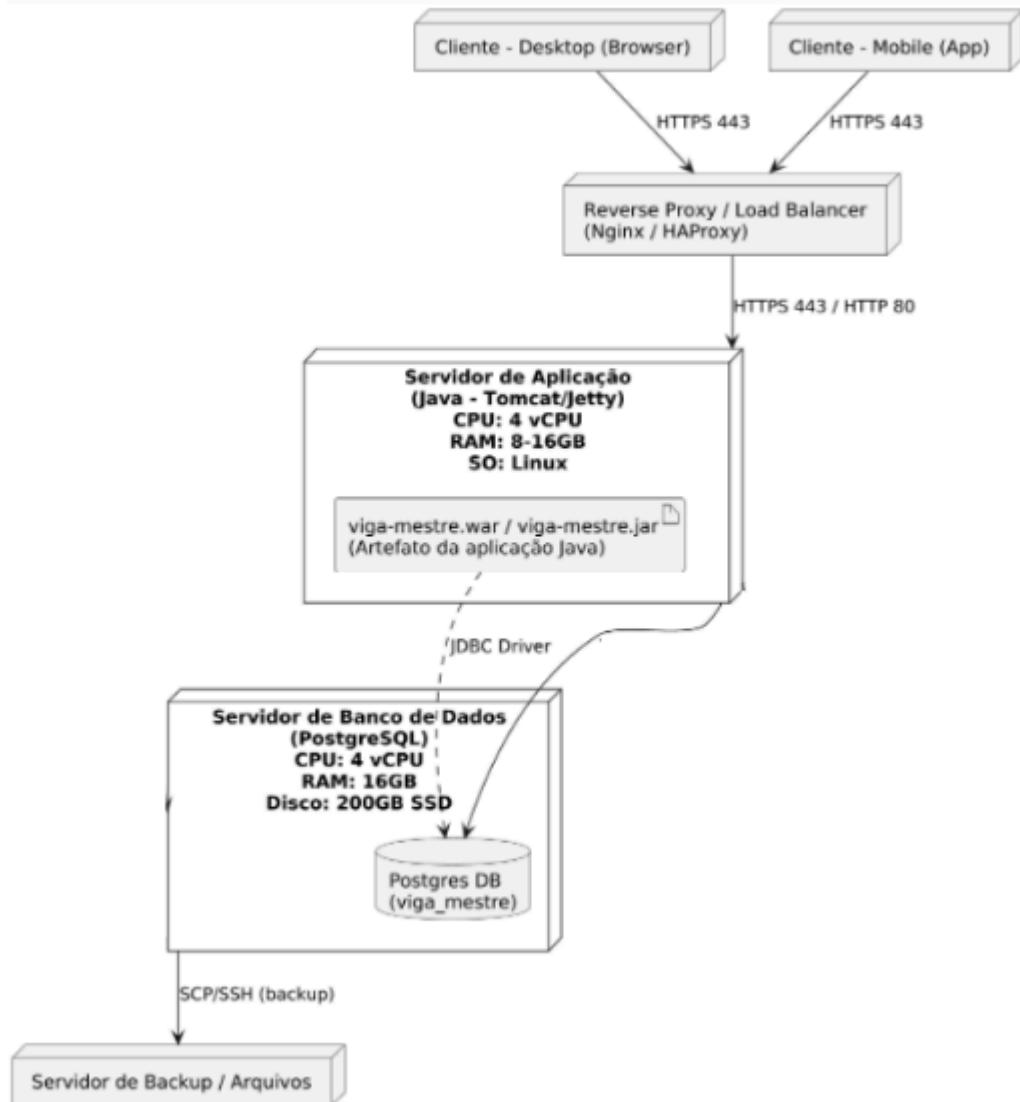
APARTAMENTO/UNIDADE



SERVICO



## DIAGRAMAS DE IMPLANTAÇÃO



## NÍVEL DE ACESSO POR TIPO DE USUÁRIO

TIPO DE USUÁRIO	PERMISSÕES CONCEDIDAS
Administrador	Acesso completo ao sistema, com permissão exclusiva para cadastrar novos usuários e realizar configurações gerais do sistema.
Gerente de Obras	Responsável pela gestão macro. Tem permissão para criar novas obras, contratar serviços e abrir frentes de trabalho (tarefas), definindo a quantidade necessária de

	funcionários para cada atividade.
Mestre de Obras	Responsável pela supervisão direta. Tem permissão para solicitar serviços/materiais e supervisionar a execução das tarefas no canteiro.
Funcionário	Usuário operacional. Tem permissão para visualizar o quadro de tarefas disponíveis e realizar a auto-atribuição (assumir uma tarefa livre), além de atualizar seus status (ex: "Em andamento", "Concluído").
Financeiro	Permissão para registrar despesas e visualizar relatórios de gastos gerados pelo módulo Financeiro da obra.
Cliente / Comprador	Perfil externo. Utiliza a plataforma para visualizar unidades disponíveis (apartamentos), realizar simulações de impostos e efetivar a compra/registro de uma unidade.

## PONTUAÇÃO DO PROJETO

### CÁLCULO DO PESO DOS ATORES

<b>Tipo de Ator</b>	<b>Descrição</b>	<b>Quantidade</b>	<b>Peso</b>	<b>Total</b>
Simples	Sistema que interage via API	0	1	0
Médio	Usuário que interage via linha de comando	0	2	0
Complexo	Usuário que interage via GUI	7	3	21
<b>Total UAW</b>				<b>21</b>

#### CÁLCULO DO PESO DOS CASOS DE USO

Os casos de uso foram classificados pela sua complexidade, medida em número de transações.

<b>Tipo de Caso de Uso</b>	<b>Transações</b>	<b>Quantidade</b>	<b>Peso</b>	<b>Total</b>
Simples	1 a 3	10	5	50
Médio	4 a 7	5	10	50
Complexo	8 ou mais	0	15	0
<b>Total UUCW</b>		<b>15</b>		<b>100</b>

#### PONTOS DE CASO DE USO NÃO AJUSTADOS

$$UUCP = UAW + UUCW$$

$$UUCP = 21 + 100 = \mathbf{121}$$

#### FATOR DE COMPLEXIDADE TÉCNICA

Fator	Descrição	Peso	Fator	Descrição	Peso
T1	Sistema distribuído	2	T8	Portabilidade	2
T2	Desempenho	4	T9	Facilidade de mudança	3
T3	Eficiência do usuário final	3	T10	Concorrência	3
T4	Processamento interno complexo	2	T11	Requisitos de segurança	3
T5	Reutilização de código	4	T12	Acesso para terceiros	0
T6	Facilidade de instalação	3	T13	Necessidade de treinamento	1
T7	Facilidade de uso	4		<b>Soma (TFactor)</b>	<b>34</b>

$$TCF = 0.6 + (0.01 * TFactor) = 0.6 + (0.01 * 34) = \mathbf{0.94}$$

## FATOR DE COMPLEXIDADE AMBIENTAL

Fator	Descrição	Peso	Fator	Descrição	Peso
F1	Familiaridade com a metodologia	3	F5	Motivação da equipe	4
F2	Experiência na aplicação	4	F6	Estabilidade dos requisitos	2
F3	Experiência em orientação a objetos	4	F7	Pessoal em tempo parcial	1
F4	Capacidade do analista líder	4	F8	Dificuldade da linguagem	2
				<b>Soma (EFactor)</b>	<b>24</b>

$$ECF = 1.4 + (-0.03 * EFactor) = 1.4 - (0.03 * 24) = \mathbf{0.68}$$

## CÁLCULO FINAL E ESTIMATIVA DE ESFORÇO

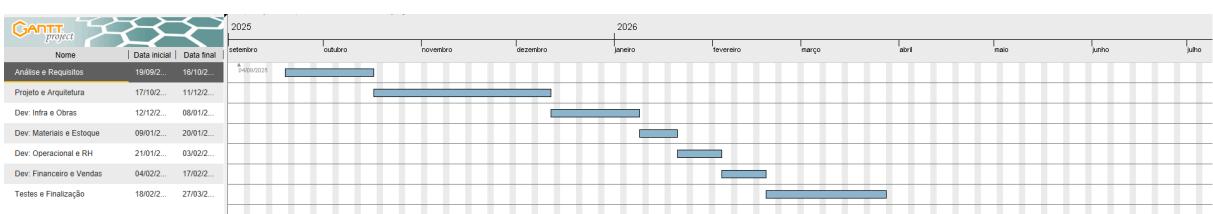
$$\text{UCP (Ajustado)} = \text{UUCP} * \text{TCF} * \text{ECF} = 121 * 0.94 * 0.68 \approx \mathbf{77,43 \text{ UCP}}$$

$$\text{Esforço Total Estimado} = 77,43 \text{ UCP} * 20 \text{ horas/UCP} = \mathbf{1.546,8 \text{ horas}}$$

## DIAGRAMA DE GANTT

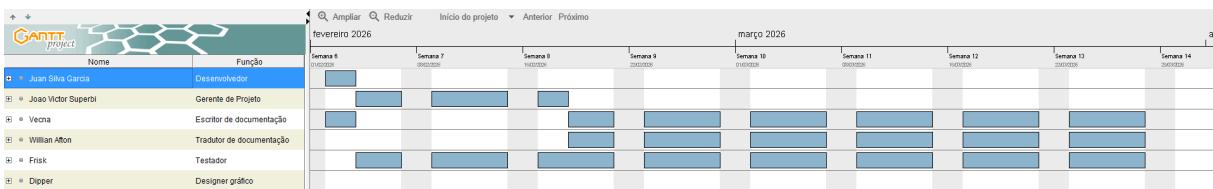
O cronograma a seguir detalha as principais fases do projeto, com base na data de início do documento e no esforço total estimado.

*Figura 10: Diagrama de Gantt(Tarefas)*



*Fonte: autor*

*Figura 11: Diagrama de Gantt(Pessoas)*



*Fonte: autor*

ID	Tarefa	Período (Sprints)	Início	Fim
1	Análise e Planejamento	Sprints 1-2	19/09/25	16/10/25
2	Projeto e Arquitetura	Sprints 3-6	17/10/25	11/12/25
3	Dev: Infra e Obras	Sprints 7-8	12/12/25	08/01/26

4	Dev: Materiais e Estoque	Sprint 9	09/01/26	20/01/26
5	Dev: Operacional e RH	Sprint 10	21/01/26	03/02/26
6	Dev: Financeiro e Vendas	Sprint 11	04/02/26	17/02/26
7	Testes e Finalização	Sprint 12	18/02/26	27/02/26

## DEFINIÇÃO DO BACKLOG E PRIMEIRO SPRINT NO SCRUM

### PRODUCT BACKLOG

O Product Backlog prioriza as funcionalidades do sistema na forma de Histórias de Usuário.

ID	História de Usuário (User Story)	Requisito(s)	Prioridade
US01	Como Administrador, quero gerenciar o cadastro de funcionários (CRUD) para manter a equipe da empresa atualizada.	RF001-RF004	Essencial
US02	Como Gerente de Obras, quero	RF005	Essencial

	gerenciar o cadastro de obras (CRUD) para organizar os projetos.		
US03	Como Gerente de Obras, quero definir o cronograma macro de uma obra para planejar as etapas principais.	RF006	Essencial
US04	Como Engenheiro, quero atribuir tarefas do cronograma a funcionários específicos para delegar responsabilidades.	RF007	Essencial
US05	Como Funcionário, quero atualizar o status das minhas tarefas para reportar o meu progresso.	RF008	Essencial
US06	Como Administrador, quero gerenciar o cadastro de materiais (CRUD) para controlar o que pode ser usado nas obras.	RF011	Essencial
US07	Como Administrador,	RF012	Importante

	quero gerenciar as categorias de materiais para organizar o estoque.		
US08	Como Administrador, quero gerenciar o cadastro de fornecedores (CRUD) para saber de quem comprar materiais.	RF013	Importante
US09	Como Financeiro, quero lançar as despesas de uma obra para manter o controle de custos.	RF014	Essencial
US10	Como Financeiro ou Gerente de Obras, quero gerar um relatório de gastos por obra para analisar o orçamento.	RF015	Essencial
US11	Como Gerente de Obras, quero gerar um relatório gerencial da obra para avaliar o progresso geral.	RF016	Importante
US12	Como Cliente, quero poder registrar a venda de um apartamento para formalizar a	RF017	Essencial

	transação.		
US13	Como Gerente de Obras, quero gerenciar os serviços contratados para uma obra.	RF010	Importante

## SPRINT

Sprint	Fase / Foco	Período	Dias	Entregáveis e atividades
1	Análise de Requisitos	19/09/25 a 01/10/25	9	<ul style="list-style-type: none"> <li>• Levantamento com cliente</li> <li>• Documento de Requisitos</li> <li>• Diagrama de Caso de Uso</li> <li>• Descrição de Casos de Uso</li> </ul>
2	Modelagem de Análise	02/10/25 a 16/10/25	10	<ul style="list-style-type: none"> <li>• Diagrama de Classe (Análise)</li> <li>• Diagrama de Atividades</li> <li>• Estimativa por Pontos de Caso de Uso</li> <li>• Diagrama de Gantt</li> <li>• Definição do Backlog e 1º Sprint</li> </ul>
3	Projeto e Prototipação	17/10/25 a 30/10/25	10	<ul style="list-style-type: none"> <li>• Gerência de Configuração (Git e GitHub)</li> <li>• Prototipação de Telas (Interfaces)</li> <li>• Diagrama de Estados</li> </ul>

4	Arquitetura (Estrutura)	31/10/25 a 13/11/25	9	<ul style="list-style-type: none"> <li>Diagrama de Classe de Projeto (aplicando padrões Singleton, DAO e MVC)</li> </ul>
5	Arquitetura (Comportamento)	14/11/25 a 27/11/25	9	<ul style="list-style-type: none"> <li>Diagrama de Sequência de Projeto (aplicando padrões Singleton, DAO e MVC)</li> <li>Diagrama de Implantação</li> </ul>
6	Revisão e Preparação	28/11/25 a 11/12/25	10	<ul style="list-style-type: none"> <li>Revisão final de todos os diagramas</li> <li>Preparação do ambiente de desenvolvimento</li> <li>Validação da arquitetura</li> </ul>
7	Dev: Infra e Acesso	12/12/25 a 23/12/25	10	<ul style="list-style-type: none"> <li>Modelagem física do Banco de Dados</li> <li>Implementação da Classe Usuário</li> <li>Tela de Login e Autenticação</li> </ul>
8	Dev: Núcleo (Obras)	26/12/25 a 08/01/26	9	<ul style="list-style-type: none"> <li>CRUD de Obras</li> <li>Cadastro de Terrenos</li> <li>Lógica de Cronograma e Estágios</li> </ul>
9	Dev: Materiais e Estoque	09/01/26 a 20/01/26	10	<ul style="list-style-type: none"> <li>CRUD de Materiais e Categorias</li> <li>Gestão de Fornecedores</li> <li>Funcionalidade de Requisitar Material</li> </ul>
10	Dev: Operacional e RH	21/01/26 a 03/02/26	10	<ul style="list-style-type: none"> <li>Cadastro de Funcionários (Pedreiro, Engenheiro, etc.)</li> </ul>

				<ul style="list-style-type: none"> <li>• Gestão e Atribuição de Tarefas</li> <li>• Atualização de status das tarefas</li> </ul>
11	Dev: Comercial e Financeiro	04/02/26 a 17/02/26	10	<ul style="list-style-type: none"> <li>• Cadastro de Apartamentos (Unidades)</li> <li>• Sistema de Vendas</li> <li>• Lançamento de Despesas e Relatórios</li> </ul>
12	Finalização	18/02/26 a 27/02/26	8	<ul style="list-style-type: none"> <li>• Testes integrados (End-to-end)</li> <li>• Correção de bugs</li> <li>• Preparação para entrega final</li> </ul>

#### SPRINT BACKLOG

ID	Item	Tarefas Associadas
US14	(Técnica) Configurar o ambiente de desenvolvimento	<ul style="list-style-type: none"> <li>- Instalar e configurar o banco de dados Postgres.</li> <li>- Criar a estrutura inicial do projeto Java.</li> <li>- Definir a arquitetura base do sistema.</li> </ul>
US01	Como Administrador, quero gerenciar o cadastro de funcionários (CRUD)	<ul style="list-style-type: none"> <li>- Desenvolver a tela de listagem e cadastro de funcionários.</li> <li>- Implementar as funcionalidades de salvar,</li> </ul>

		<p>editar e excluir no backend.</p> <ul style="list-style-type: none"> <li>- Criar os testes unitários para o CRUD de funcionários.</li> </ul>
<b>US02</b>	Como Gerente de Obras, quero gerenciar o cadastro de obras (CRUD)	<ul style="list-style-type: none"> <li>- Desenvolver a tela de listagem e cadastro de obras.</li> <li>- Implementar as funcionalidades de salvar, editar e excluir no backend.</li> <li>- Criar os testes unitários para o CRUD de obras.</li> </ul>