
Documentação de Projeto

para o sistema

Domus Finanças

Versão 1.0

Projeto de sistema elaborado pelos alunos Davi Érico dos Santos e Lucas Maia Rocha
como parte da disciplina **Projeto de Software**.

28/05/2025

Tabela de Conteúdo

1. Introdução	1
2. Modelos de Usuário e Requisitos	1
2.1 Descrição de Atores	1
2.2 Modelo de Casos de Uso e Histórias de Usuários	1
2.3 Diagrama de Sequência do Sistema e Contrato de Operações	1
3. Modelos de Projeto	1
3.1 Arquitetura	1
3.2 Diagrama de Componentes e Implantação.	2
3.3 Diagrama de Classes	2
3.4 Diagramas de Sequência	2
3.5 Diagramas de Comunicação	2
3.6 Diagramas de Estados	2
4. Modelos de Dados	2

Histórico de Revisões

Nome	Data	Razões para Mudança	Versão
Lucas Maia	28/05	Melhoras nos diagramas	1.0
Davi Érico	29/05	Melhoras nos diagramas	1.1

1. Introdução

Este documento apresenta a documentação de projeto do sistema **Domus Finanças**, uma aplicação voltada para o controle financeiro pessoal, cujo objetivo é auxiliar usuários no gerenciamento de suas finanças de maneira prática, intuitiva e segura. O sistema oferece funcionalidades como cadastro de receitas e despesas, categorização de transações, geração de relatórios financeiros e acompanhamento do saldo mensal, promovendo maior organização e planejamento econômico.

A elaboração deste projeto contempla duas principais etapas:

1. A definição e revisão de modelos de domínio, incluindo a análise de requisitos, identificação de atores e elaboração de casos de uso.
2. A modelagem da solução, por meio da especificação da arquitetura do sistema, diagramas de interação, estrutura de dados e contratos de operação.

A referência principal para o entendimento do problema e das necessidades do usuário é o documento de especificação que descreve a visão de domínio do sistema, servindo como base para as decisões técnicas e funcionais tomadas ao longo do desenvolvimento deste projeto.

2. Modelos de Usuário e Requisitos

2.1 Descrição de Atores

Usuário Comum

É o principal ator do sistema. Representa qualquer pessoa que deseja organizar suas finanças pessoais. O usuário pode cadastrar e gerenciar receitas e despesas, visualizar relatórios financeiros, definir metas de economia, e acompanhar seu saldo mensal. Todo o acesso ao sistema ocorre por meio de autenticação via login e senha.

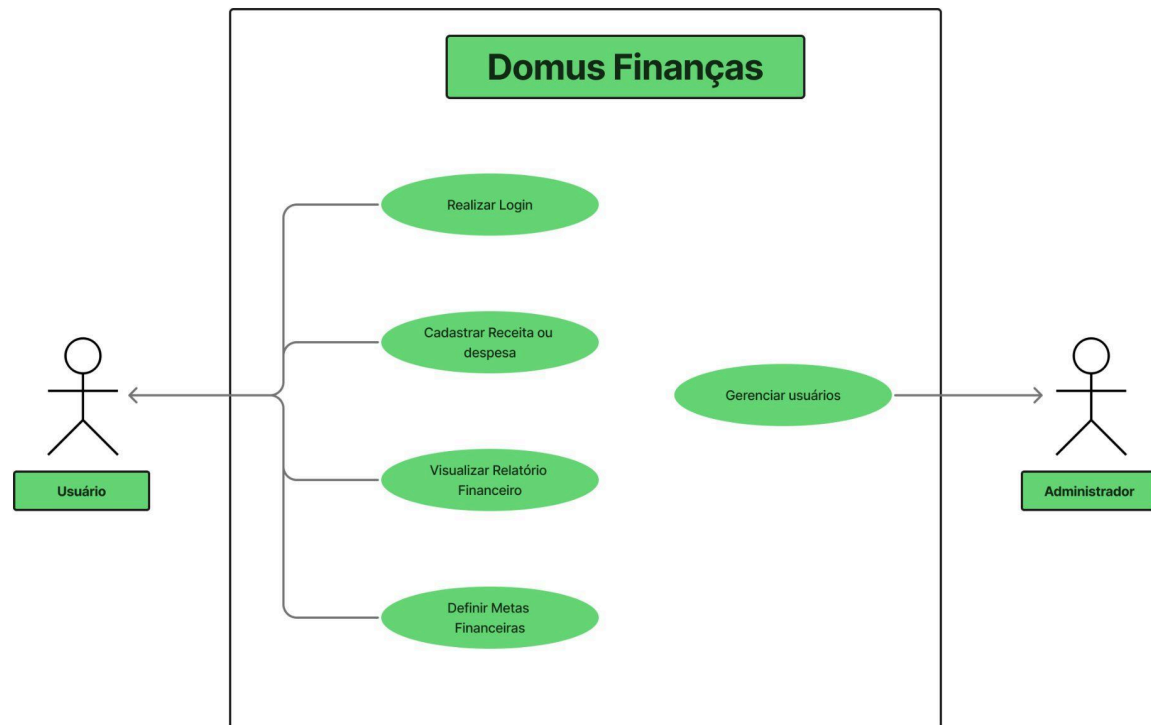
Administrador do Sistema

Responsável pela manutenção geral da aplicação. Tem acesso a funcionalidades como o gerenciamento de usuários (bloquear ou excluir contas), análise de logs de uso e supervisão do desempenho do sistema. Este ator não interage com o controle financeiro em si, mas garante o funcionamento e a segurança da plataforma.

2.2 Modelo de Casos de Uso

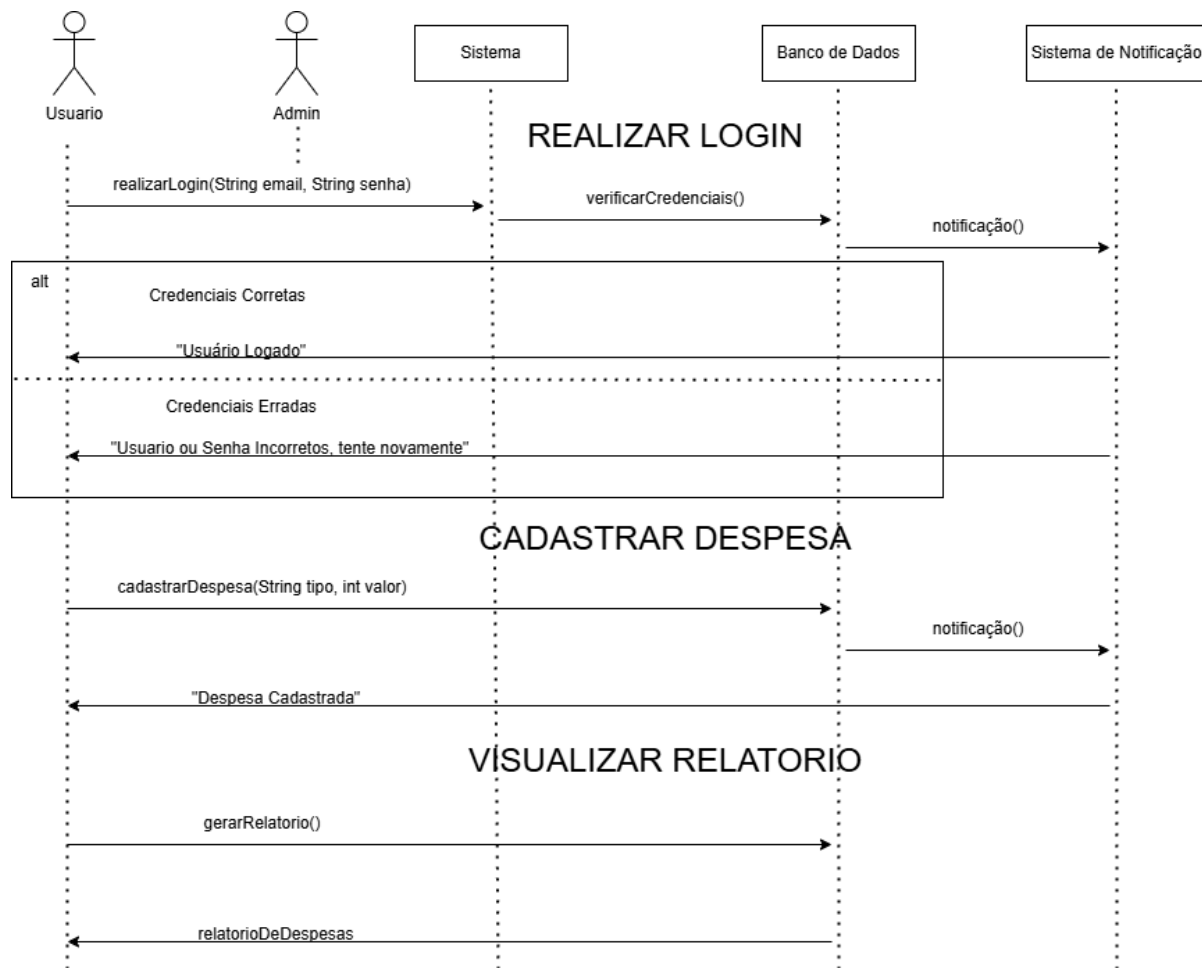
Abaixo são descritos os principais casos de uso do sistema **Domus Finanças**, que representam as interações entre os atores e as funcionalidades oferecidas pela aplicação.

ID	Nome do Caso de Uso	Atores Envolvidos	Descrição Resumida
UC-01	Realizar Login	Usuário, Administrador	Permite que o usuário acesse o sistema através de login com e-mail e senha.
UC-02	Cadastrar Receita ou Despesa	Usuário	Permite o registro de entradas e saídas financeiras com categoria, data e valor.
UC-03	Visualizar Relatório Financeiro	Usuário	Gera um resumo gráfico e textual das movimentações e saldo mensal do usuário.
UC-04	Definir Metas Financeiras	Usuário	Permite que o usuário configure metas de economia mensal e acompanhe seu progresso.
UC-05	Gerenciar Usuários	Administrador	Permite ao administrador visualizar, bloquear ou excluir contas de usuários.



2.3 Diagrama de Sequência do Sistema

Nesta subseção é apresentado o diagrama de sequência do sistema de pelo menos, 3 Casos de Uso ou Histórias de Usuário descritos na Seção 2.3.



Formato para cada contrato de operação

REALIZAR LOGIN

Contrato	Usuario
Operação	realizarLogin()
Referências cruzadas	2.3 Diagrama de Sequência do Sistema - Imagem
Pré-condições	usuario <> null / usuario.email = true and usuario.senha = true
Pós-condições	O usuário consegue acessar o sistema com sucesso

CADASTRAR DESPESA

Contrato	Despesa
Operação	cadastrarDespesa(String tipo, int valor)
Referências cruzadas	2.3 Diagrama de Sequência do Sistema - Imagem
Pré-condições	despesa.tipo <> null and despesa.valor <> null / usuario = logado
Pós-condições	O usuário cadastra uma despesa no sistema

VISUALIZAR RELATORIO

Contrato	Relatorio
Operação	visualizarRelatorioDespesas()
Referências cruzadas	2.3 Diagrama de Sequência do Sistema - Imagem
Pré-condições	despesas.tipo > 0 / usuario = logado
Pós-condições	O usuário consegue visualizar um relatório coerente com suas despesas cadastradas

3. Modelos de Projeto

3.1 Arquitetura

3.1 Arquitetura

A arquitetura do sistema **Domus Finanças** adota um modelo em camadas, seguindo a estrutura clássica de aplicações baseadas em web. Essa separação facilita a manutenção, escalabilidade e organização do projeto. O sistema será composto por quatro camadas principais:

1. Camada de Apresentação (Frontend)

Responsável pela interface com o usuário. Aqui são desenvolvidas as páginas e componentes gráficos que possibilitam a interação com as funcionalidades do sistema. Esta camada se comunica diretamente com a camada de aplicação por meio de requisições HTTP.

- Tecnologias: HTML, CSS, JavaScript

2. Camada de Aplicação (Backend)

Contém a lógica de negócio do sistema. É responsável por processar os dados recebidos da camada de apresentação, aplicar regras e encaminhar respostas. Também gerencia o controle de sessões, autenticação e autorização.

- Tecnologias: Java com Spring Boot.

3. Camada de Persistência (DAO)

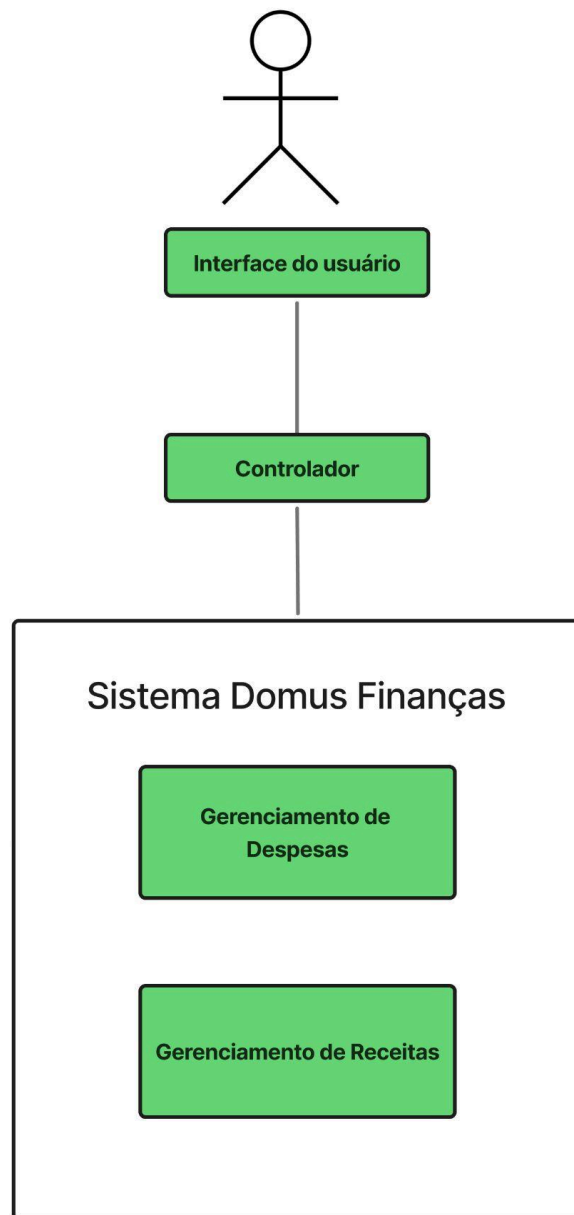
Gerencia o acesso aos dados armazenados no banco. Faz a mediação entre a aplicação e o banco de dados, utilizando padrões como DAO (Data Access Object) ou JPA (se Java for usado).

- Operações: CRUD de receitas, despesas, metas e usuários.

4. Camada de Dados (Banco de Dados)

Armazena todas as informações estruturadas do sistema, como registros de transações, usuários e configurações de metas financeiras.

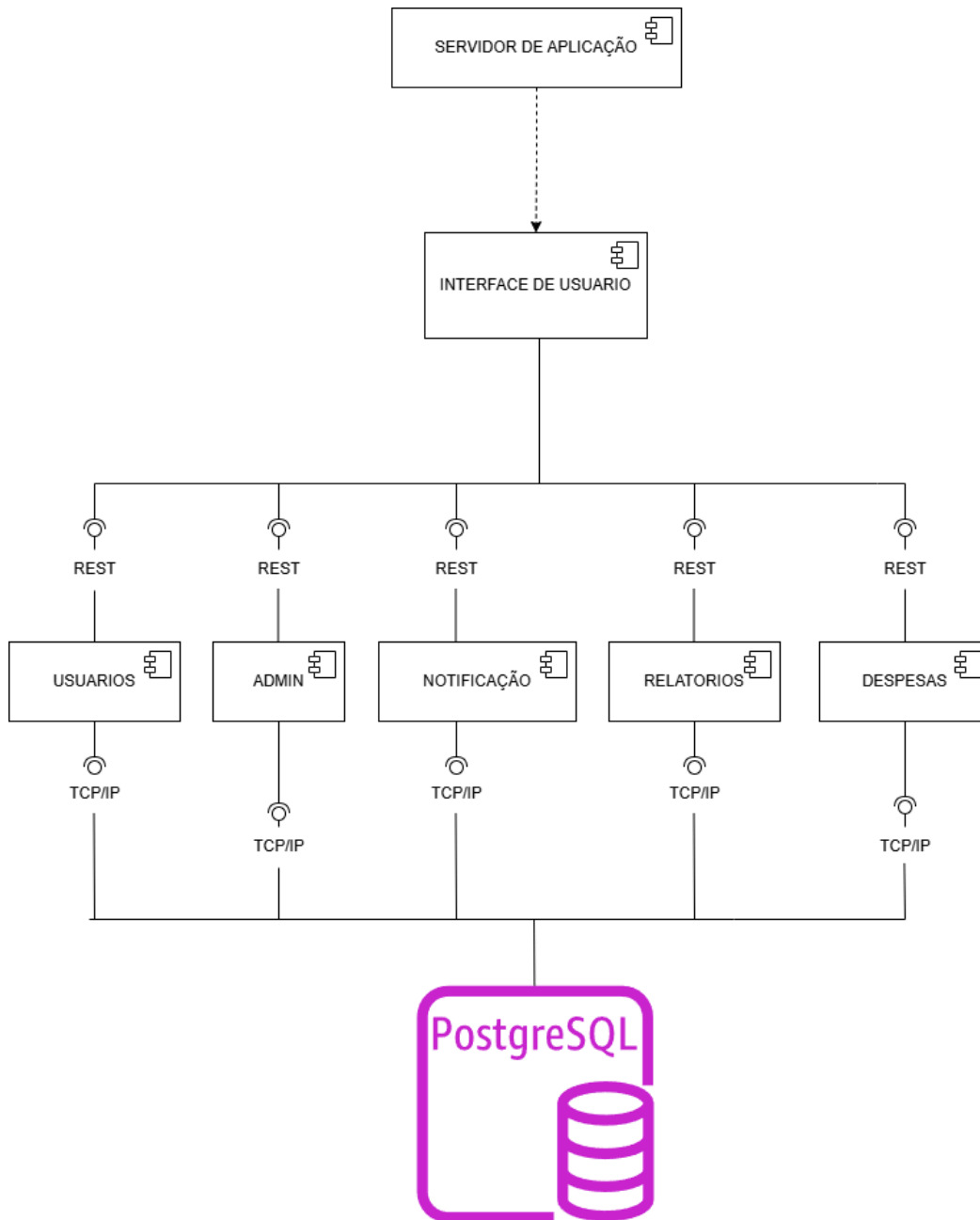
- Banco: PostgreSQL



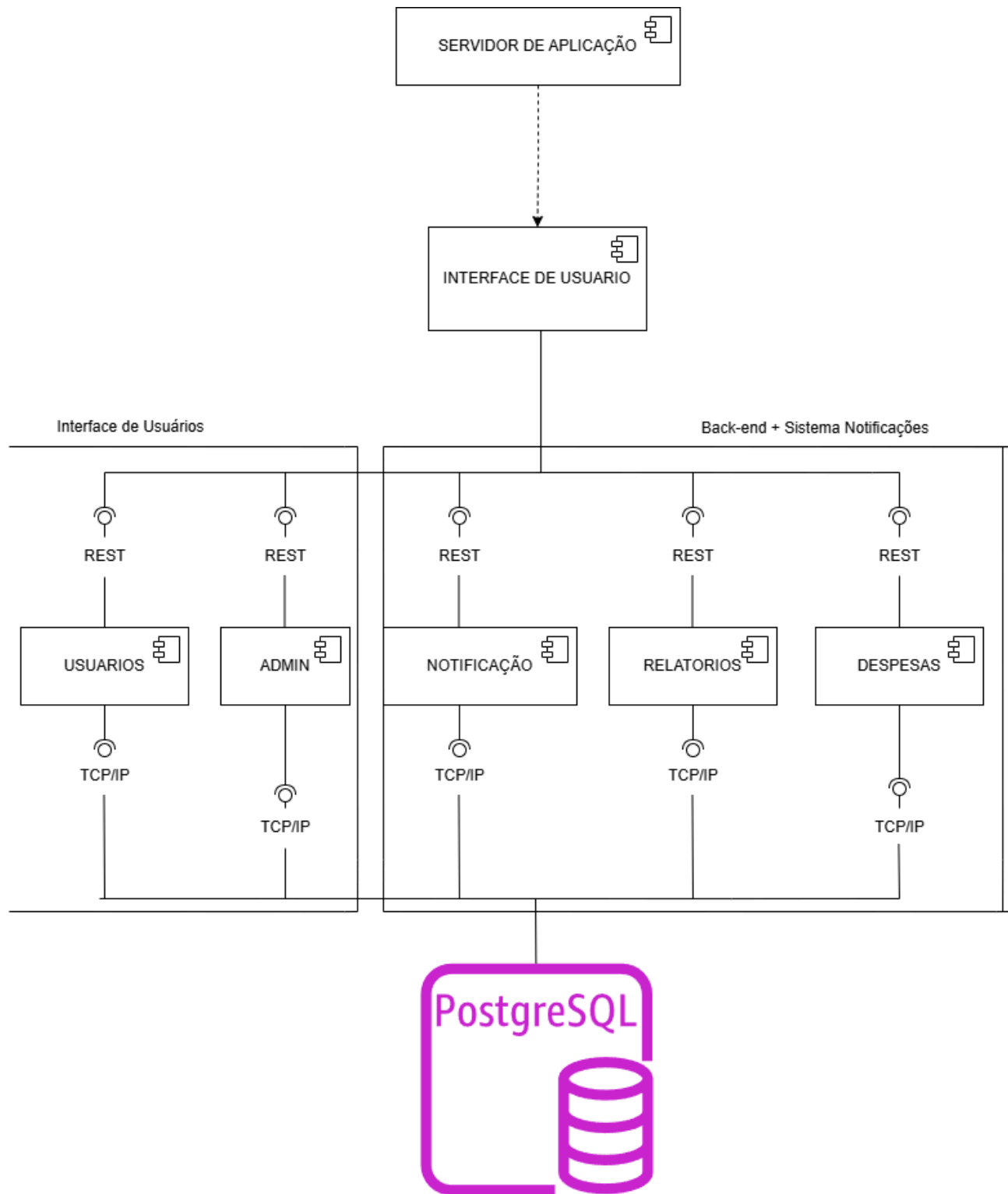
3.2 Diagrama de Componentes e Implantação.

Diagramas de componentes do sistema. Diagrama de implantação mostrando onde os componentes estarão alocados para a execução.

Componentes



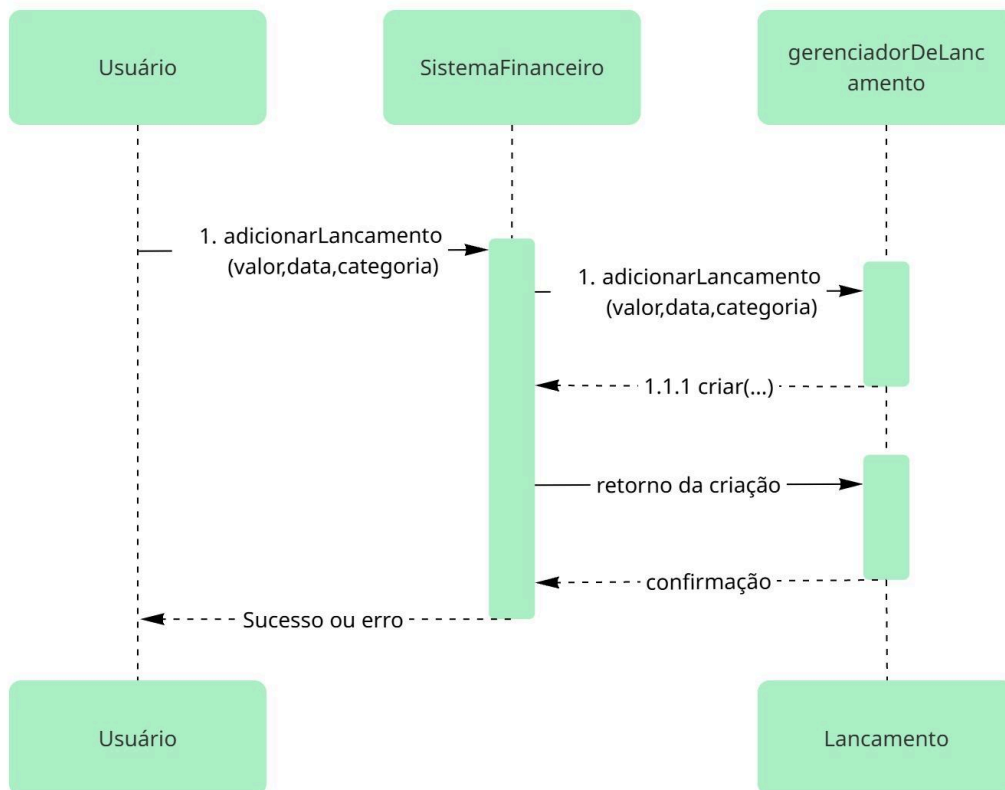
Implantação



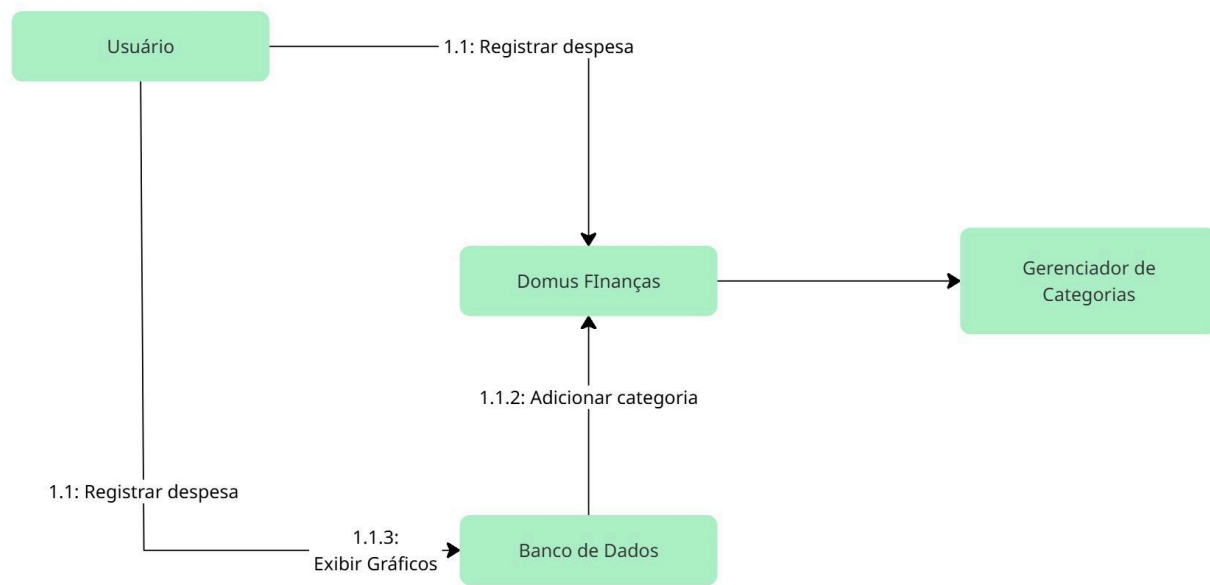
3.3 Diagrama de Classes



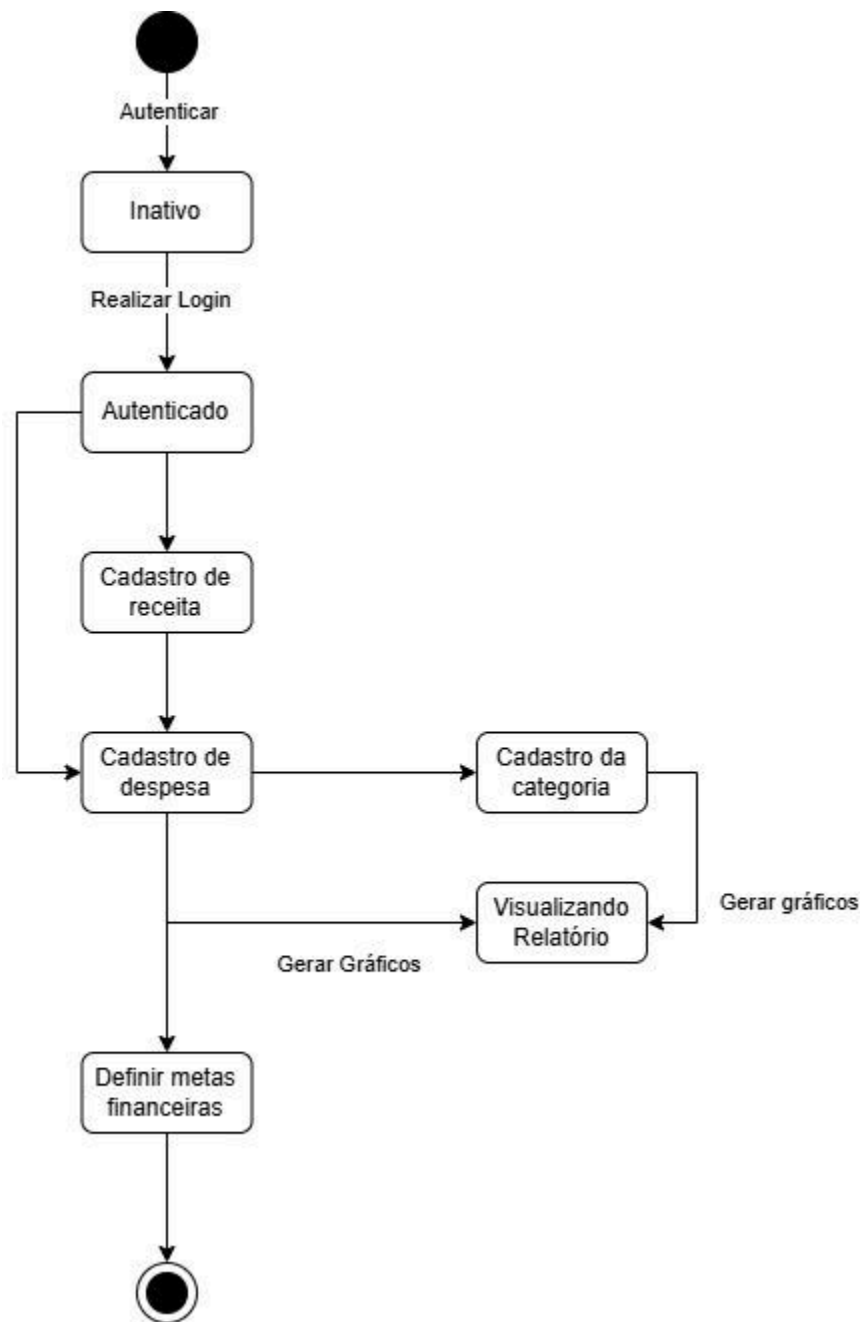
3.4 Diagramas de Sequência



3.5 Diagramas de Comunicação



3.6 Diagramas de Estados



4. Modelos de Dados

Deve-se apresentar os esquemas de banco de dados e as estratégias de mapeamento entre as representações de objetos e não-objetos.

