

# Documentação de Projeto

Projeto de sistema elaborado pelo(s) aluno(s) Arthur Freitas Jardim, Thiago Andrade Ramalho, Wilken Henrique Moreira como parte da disciplina Projeto de Software.

# README

**VERSÃO 1.0**

**11 de novembro de 2024**

## Visão geral

O **README** foi desenvolvido para oferecer uma plataforma completa que permite aos usuários gerenciar suas coleções de livros e interagir com suas leituras de forma dinâmica. A aplicação oferece uma experiência personalizada, onde tanto leitores quanto autores têm funcionalidades específicas para otimizar a gestão de livros e o reconhecimento de preferências literárias.

O sistema é estruturado em diferentes camadas de interação de acordo com os perfis de usuário. Os **leitores** podem logar com suas credenciais e gerenciar sua biblioteca pessoal, adicionar ou remover livros, e visualizar suas coleções. Além disso, eles têm a possibilidade de avaliar livros e autores, seguir autores de interesse e marcar livros como lidos, desejados ou favoritos. Por outro lado, os **Autores** têm a capacidade de gerenciar suas próprias obras, criando, atualizando e deletando livros. O sistema também inclui um **Notificador** que avisa os leitores seguidores sobre a criação de novos livros dos autores que acompanham, promovendo uma experiência de engajamento e atualização.

## Objetivos

O objetivo principal do sistema é oferecer uma plataforma intuitiva e eficiente para leitores e autores gerenciarem suas bibliotecas e interações literárias. O sistema visa permitir que leitores organizem suas coleções de livros, avaliem e sigam autores, e marquem seus livros favoritos, enquanto proporciona aos autores a capacidade de gerenciar suas obras e manter seus leitores informados sobre novos lançamentos.

### Objetivos Secundários:

1. **Acessar o sistema:** Permitir que usuários façam login com suas credenciais para utilizar a plataforma.
2. **Organizar a biblioteca:** Oferecer aos leitores a possibilidade de adicionar, remover e visualizar livros em suas coleções.
3. **Avaliar livros:** Permitir que leitores atribuam notas aos livros que leram.
4. **Seguir autores:** Oferecer aos leitores a funcionalidade de seguir autores de seu interesse.
5. **Avaliar autores:** Permitir que leitores avaliem autores com base na qualidade de seus livros.
6. **Marcar livros:** Proporcionar aos leitores a opção de marcar livros como lidos, desejados ou favoritos.
7. **Gerenciar livros:** Permitir que autores criem, atualizem e removam seus livros.
8. **Notificar leitores:** Informar os leitores que seguem um autor sobre a criação de novos livros por esse autor.

## Especificações

### 1. Cadastro de Usuários

Os usuários devem se cadastrar no sistema com informações específicas. Sendo necessário informar o nome, e-mail, e a senha.

### 2. Autenticação de Usuário

Permite que os usuários façam login com credenciais seguras, garantindo a proteção das informações com armazenamento de senhas criptografadas e sessões que mantêm o login ativo.

### 3. Gerenciamento de Biblioteca

Os leitores têm a capacidade de adicionar, remover e visualizar livros em suas coleções. A plataforma oferece filtros e funcionalidades de busca para facilitar a navegação e a organização dos títulos.

### 4. Avaliação de Livros

Os leitores podem atribuir notas e comentários aos livros que leram, com um histórico de todas as avaliações realizadas, permitindo um feedback enriquecedor sobre os títulos.

### 5. Seguir Autores

Os leitores podem seguir seus autores favoritos para receber notificações e atualizações sobre novos lançamentos, facilitando a descoberta de obras recentes.

## 6. Avaliação de Autores

Os leitores podem avaliar autores com base na qualidade de suas obras, com a visualização de notas médias e comentários, ajudando na escolha de novos livros.

## 7. Marcar Livros

Os leitores podem marcar livros como lidos, desejados ou favoritos. A plataforma oferece opções de filtros para visualizar livros por status, ajudando na organização e planejamento de leitura.

## 8. Gerenciamento de Livros (Autor)

Os autores têm uma interface dedicada para criar, atualizar e remover livros, com campos para informações como título, descrição, capa e outros detalhes importantes.

## 9. Notificação de Novos Livros (Notificador)

Quando um autor publicado cria um novo livro, os leitores que o seguem recebem notificações por meio de alertas na plataforma ou e-mails, garantindo que fiquem informados de novas publicações.

## 10. Interface de Usuário

A plataforma possui um design responsivo e amigável, projetado para oferecer uma experiência de navegação intuitiva e adaptada a diferentes dispositivos, garantindo usabilidade e acessibilidade.

## 11. Segurança e Privacidade

A proteção dos dados dos usuários é garantida com criptografia avançada e medidas contra ataques como SQL injection e XSS. O sistema também implementa níveis de autenticação para controlar o acesso e proteger as informações.

## 12. Relatórios e Estatísticas

A plataforma disponibiliza relatórios detalhados e estatísticas sobre o desempenho dos livros e interações dos usuários, com métricas de leitura e avaliação que auxiliam na análise de dados.

## 13. Tecnologia e Infraestrutura

- **Frontend:** Utilização do Vite para desenvolvimento rápido, React para uma estrutura modular e “componentizada”, e MUI React para componentes pré-prontos, com design consistente e responsivo, garantindo uma interface elegante e funcional.
- **Backend:** Node.js é utilizado para a lógica do servidor, Prisma como ORM para facilitar a interação com o banco de dados, e MySQL para o armazenamento e gerenciamento de dados, formando uma base robusta e eficiente para a aplicação.

## Tabela de Conteúdo

<b>1. Introdução</b>	<b>1</b>
<b>2. Modelos de Usuário e Requisitos</b>	<b>1</b>
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
<b>3. Modelos de Projeto</b>	<b>1</b>
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
<b>4. Modelos de Dados</b>	<b>2</b>

## Histórico de Revisões

Nome	Data	Razões para Mudança	Versão

# 1. Introdução

Este documento agrega: 1) a elaboração e revisão de modelos de domínio e 2) modelos de projeto para o sistema **README**. A referência principal para a descrição geral do problema, domínio e requisitos do sistema é o documento de especificação que descreve a visão de domínio do sistema.

## 2. Modelos de Usuário e Requisitos

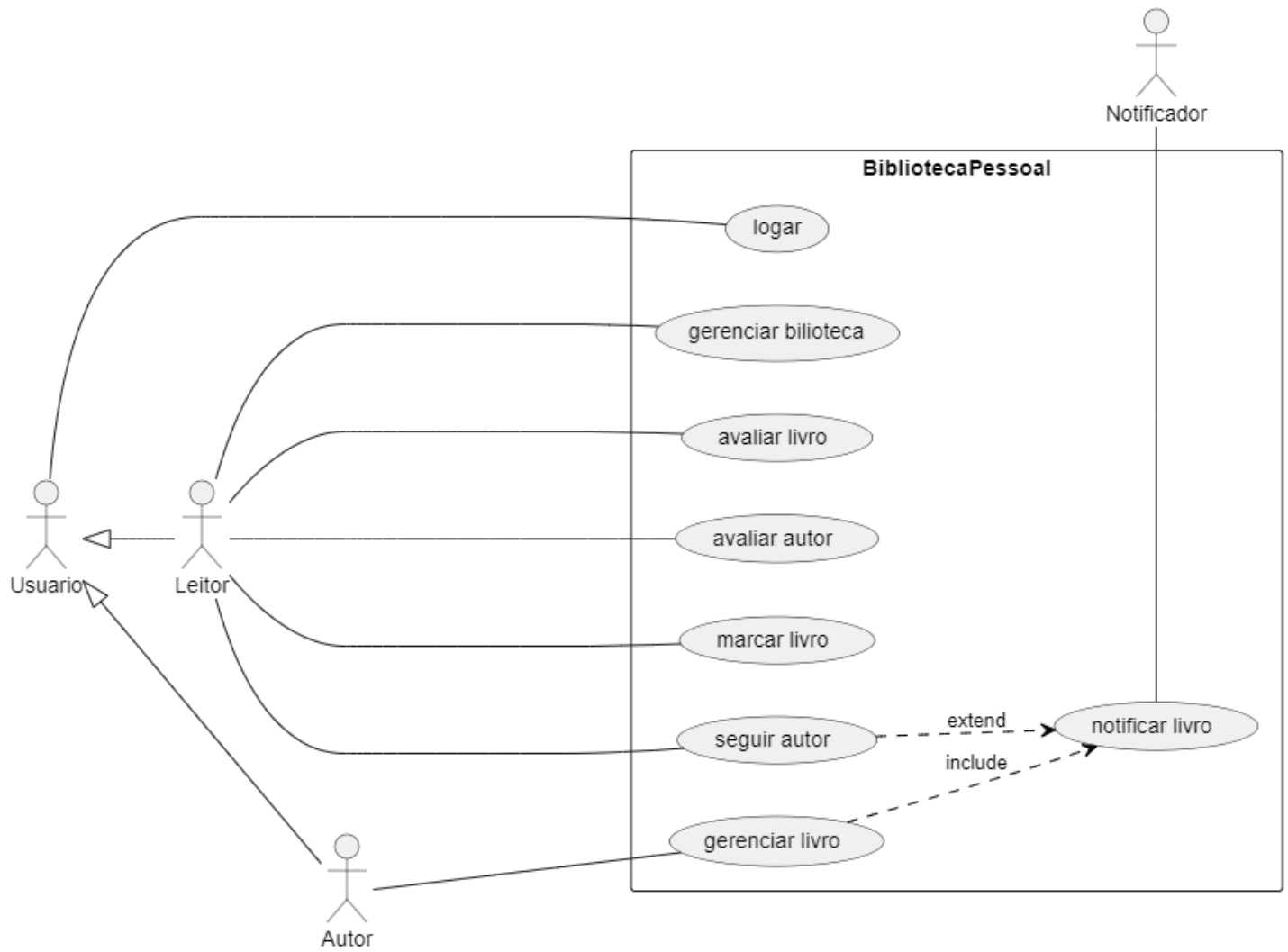
### 2.1 Descrição de Atores

O ator "Usuário" é a entidade central do sistema, responsável por iniciar a interação com a plataforma. Ele utiliza o "Sistema de Login" para realizar a autenticação com suas credenciais. Após a autenticação bem-sucedida, o usuário tem acesso às funcionalidades do sistema, podendo explorar as diferentes seções, seja como um Leitor ou como um Autor.

O "Leitor" é um ator fundamental que interage diretamente com a "Biblioteca", onde pode adicionar, remover ou visualizar livros. Ele também pode utilizar o "Sistema de Avaliação de Livros" para dar notas e fazer comentários sobre os livros que leu, além de seguir autores por meio do "Sistema de Seguimento de Autores". Com isso, o Leitor mantém uma conexão ativa com suas leituras favoritas, marcando livros como lidos, desejados ou favoritos. O "Leitor" é ainda notificado pelo "Sistema de Notificações" sobre novos lançamentos e atualizações de seus autores preferidos, garantindo que esteja sempre atualizado sobre o que há de novo em sua biblioteca.

O "Autor" é o responsável pela criação, atualização e remoção de seus próprios livros através do "Gerenciador de Livros do Autor". Ele tem o papel de enriquecer a plataforma com novas obras e manter o controle sobre seu portfólio. Quando um novo livro é publicado ou atualizado, o "Gerenciador de Livros do Autor" comunica o "Sistema de Notificações", que envia alertas para os "Leitores" que seguem o autor. Isso aumenta o engajamento e a interação entre autores e leitores, mantendo o sistema dinâmico e sempre com novos conteúdos. O "Notificador" é o ator que gerencia o envio de alertas e mensagens importantes aos leitores, mantendo-os informados sobre os lançamentos e eventos relevantes no sistema.

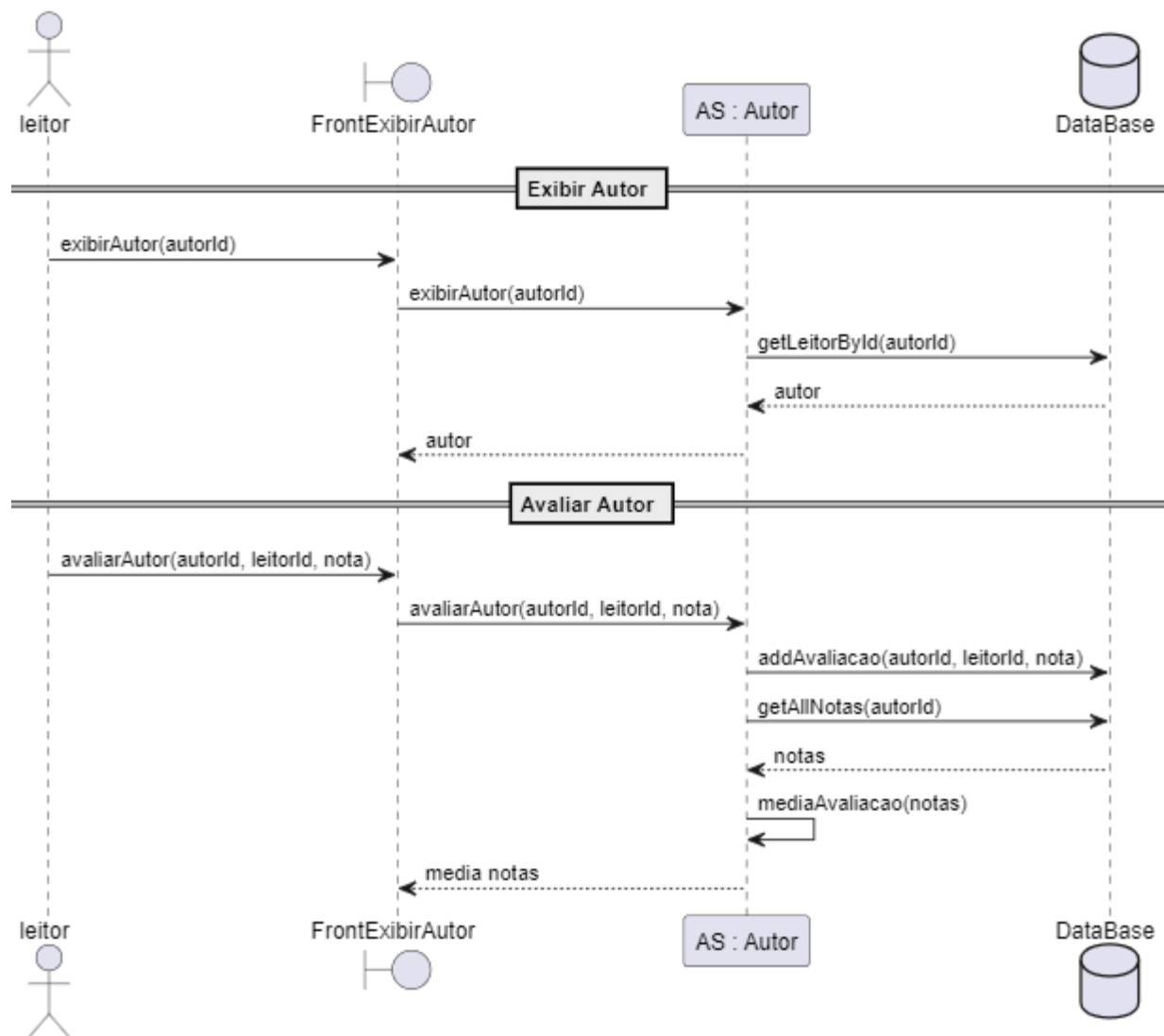
## 2.2 Modelo de Casos de Uso



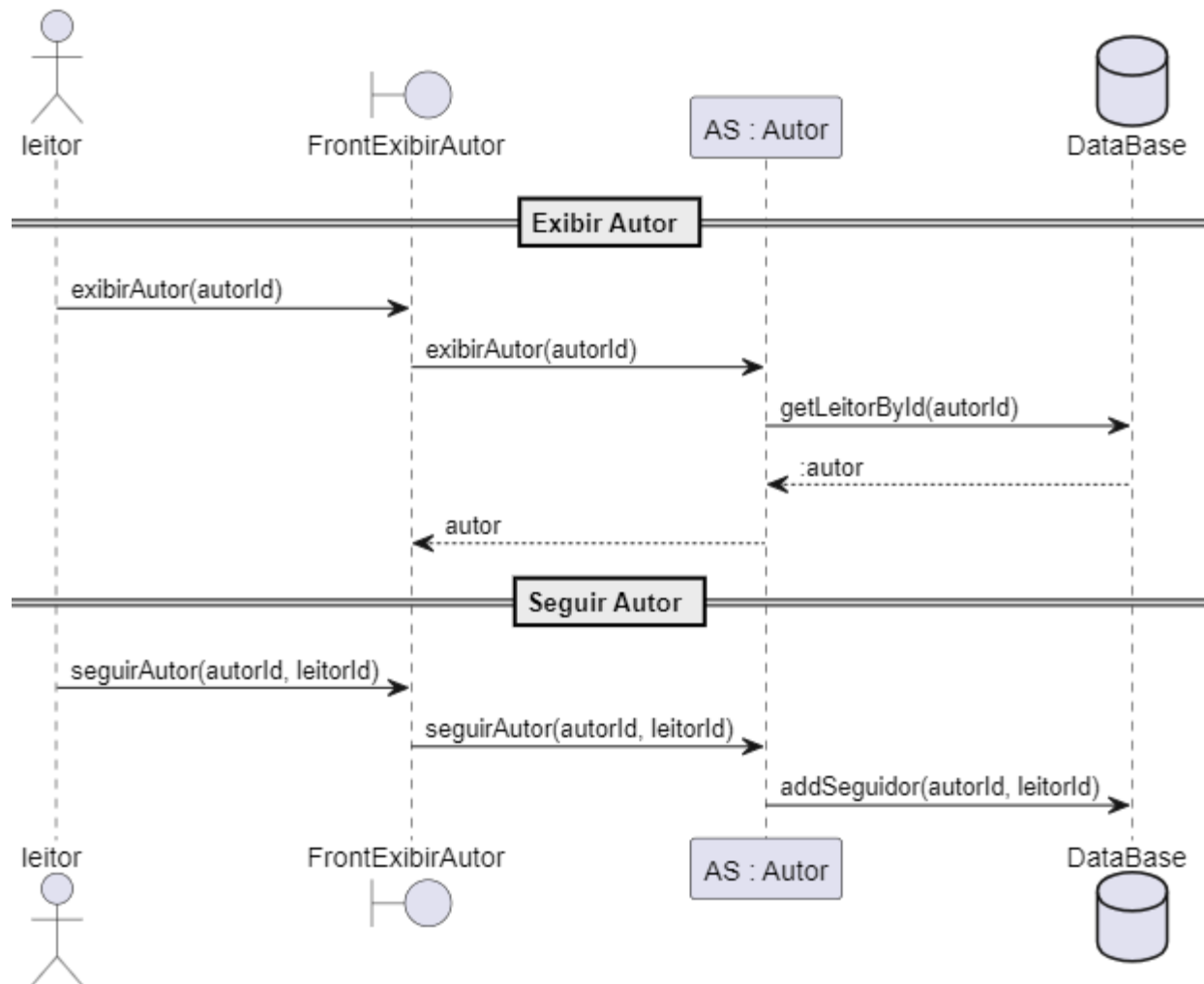


## 2.3 Diagrama de Sequência do Sistema

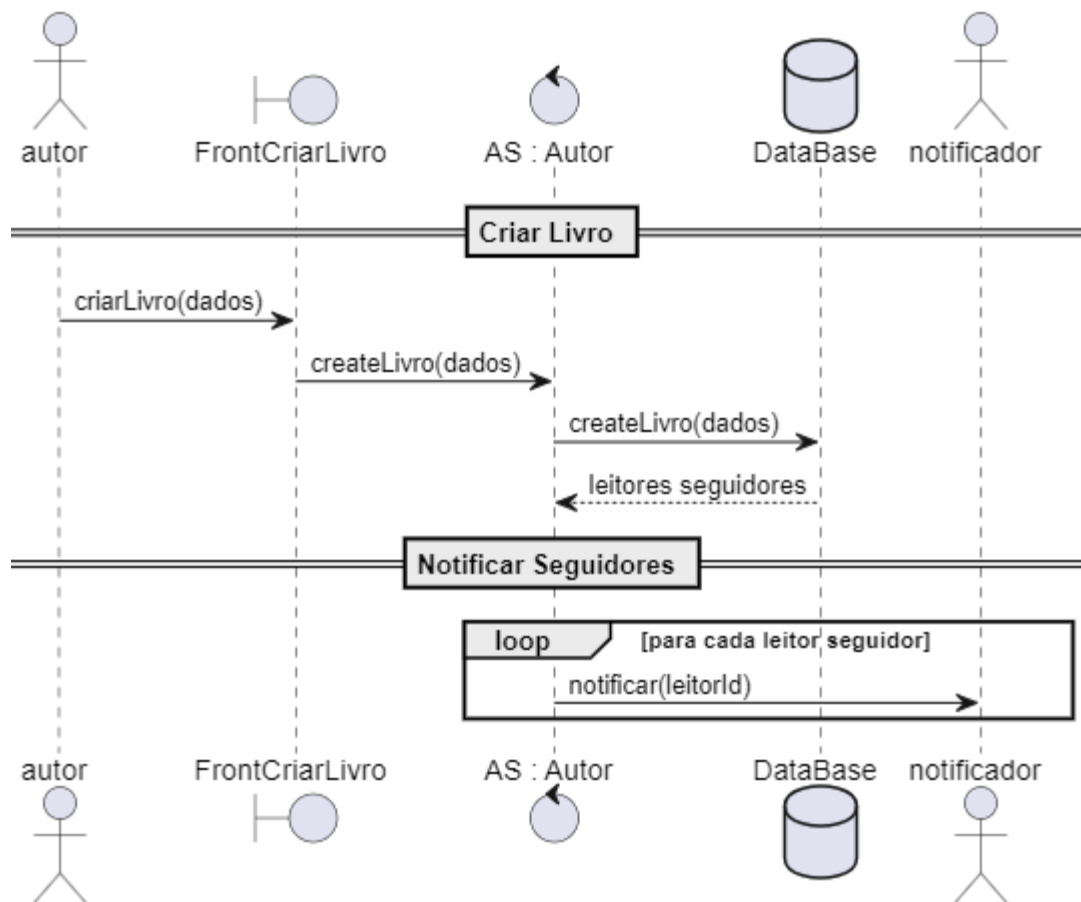
### 2.3.1 Diagrama de Sequência - UC4 Avaliar Autor



### 2.3.2 Diagrama de Sequência - UC5 Seguir Autor



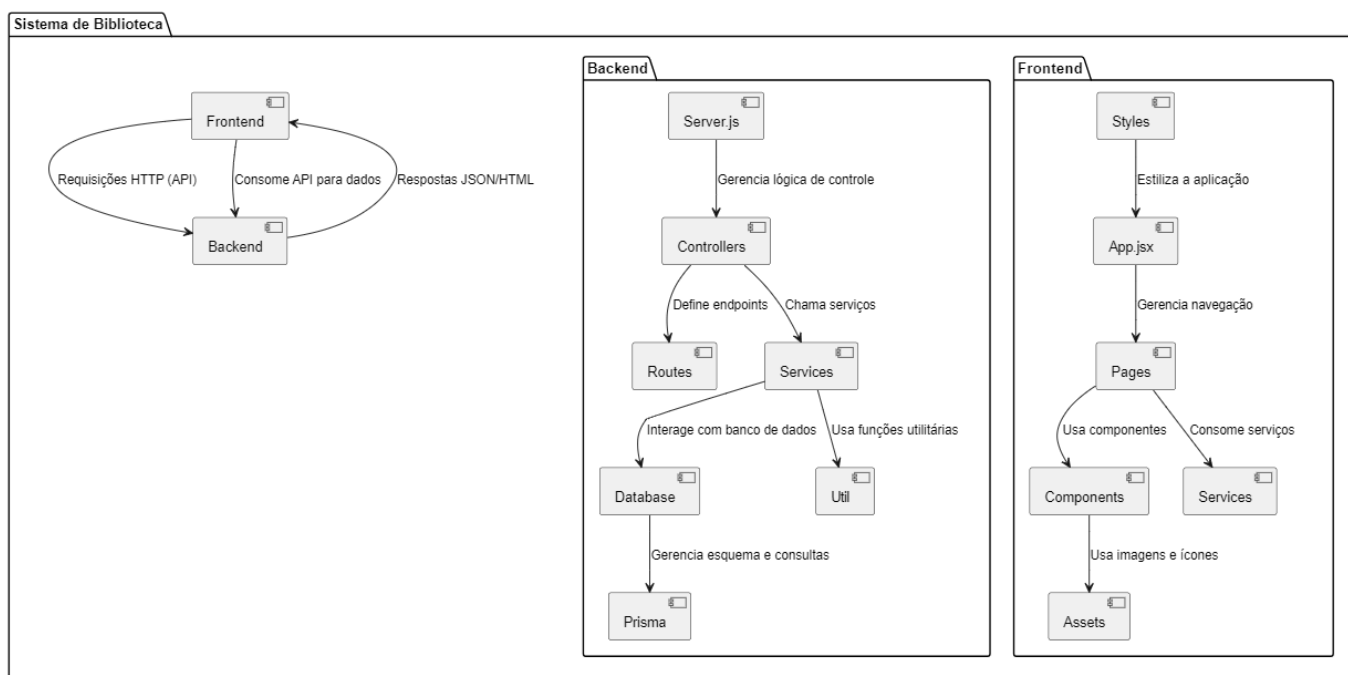
### 2.3.2 Diagrama de Sequência - UC7 Notificar Livro



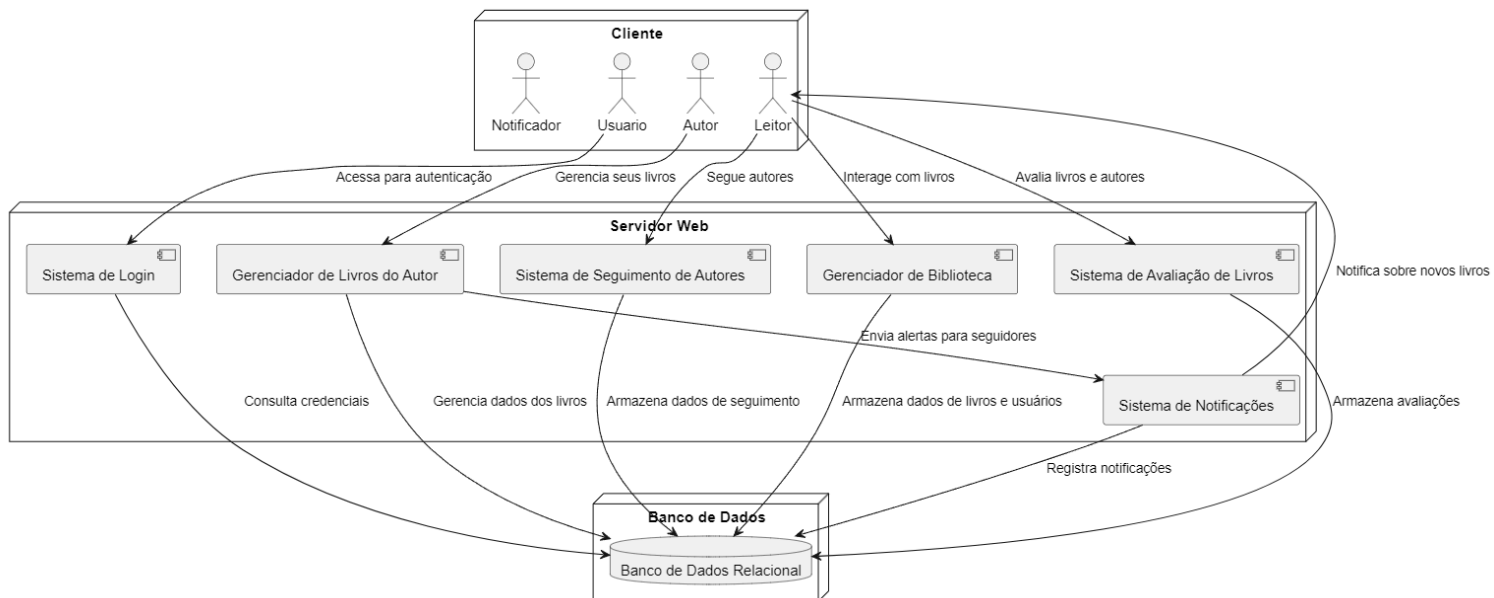
## 3. Modelos de Projeto

### 3.1 Arquitetura

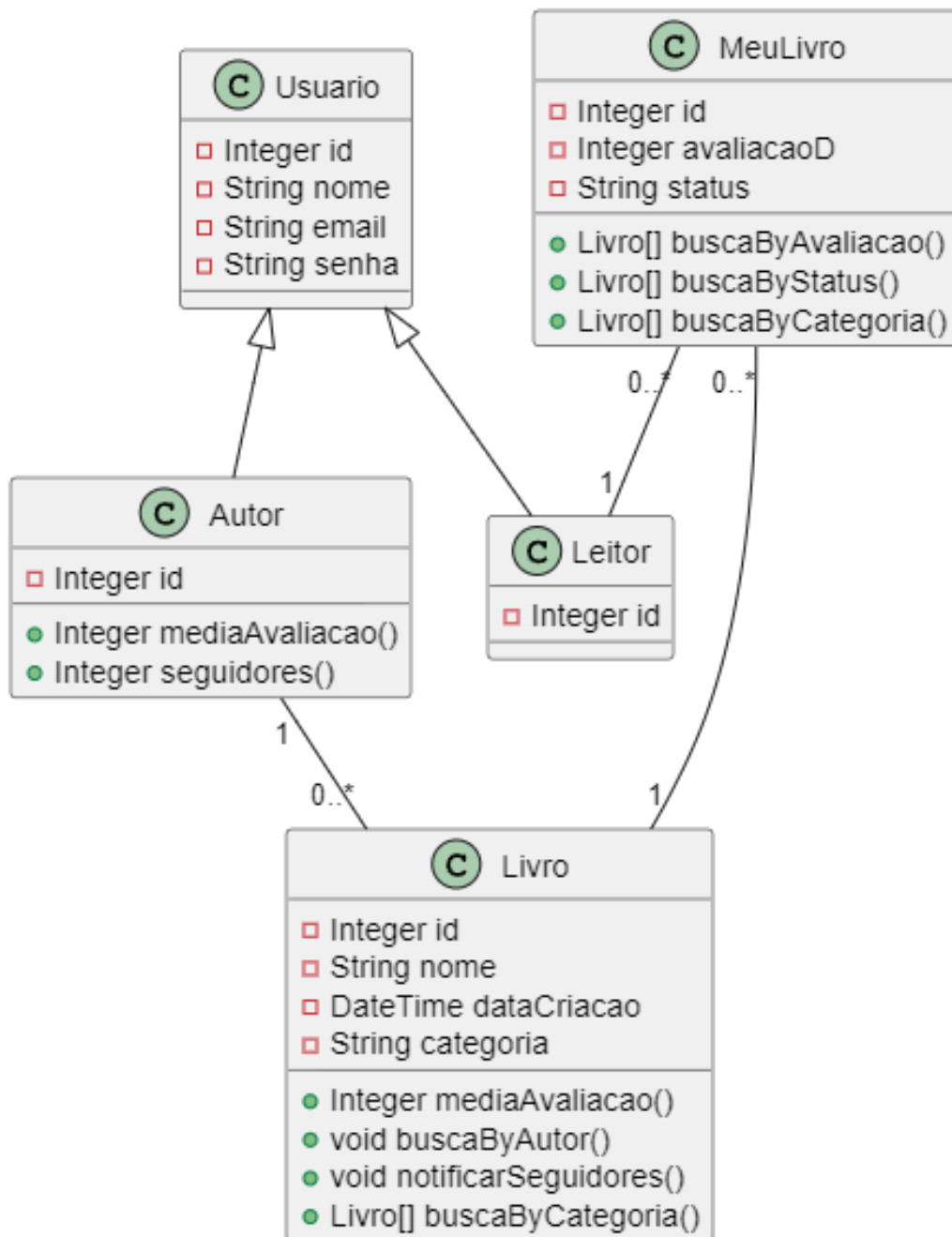
A arquitetura do sistema de biblioteca foi projetada para separar claramente as camadas frontend e backend, garantindo uma estrutura modular e escalável. O **frontend** é desenvolvido com React e Vite, proporcionando uma interface responsiva e interativa. Ele inclui componentes reutilizáveis, serviços que consomem a API do backend e folhas de estilo para uma apresentação uniforme. O **backend**, por sua vez, é implementado com Node.js e usa Prisma como ORM para gerenciar o banco de dados MySQL. Ele é composto por controladores que gerenciam a lógica de negócio, serviços que realizam operações específicas, rotas que definem os endpoints e um utilitário para funções auxiliares. A comunicação entre frontend e backend é feita via APIs REST, assegurando a troca eficiente de dados entre cliente e servidor. Essa arquitetura permite manutenção simplificada, escalabilidade e adaptação às futuras necessidades do sistema.



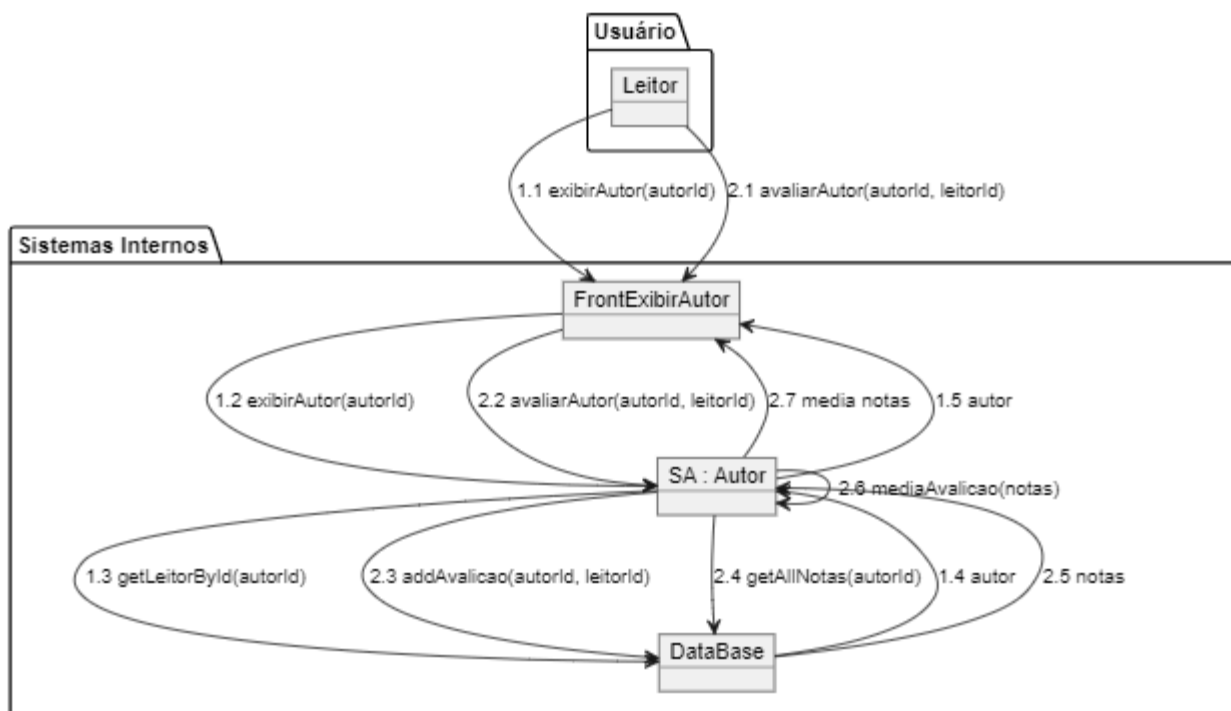
### 3.2 Diagrama de Componentes e Implantação.



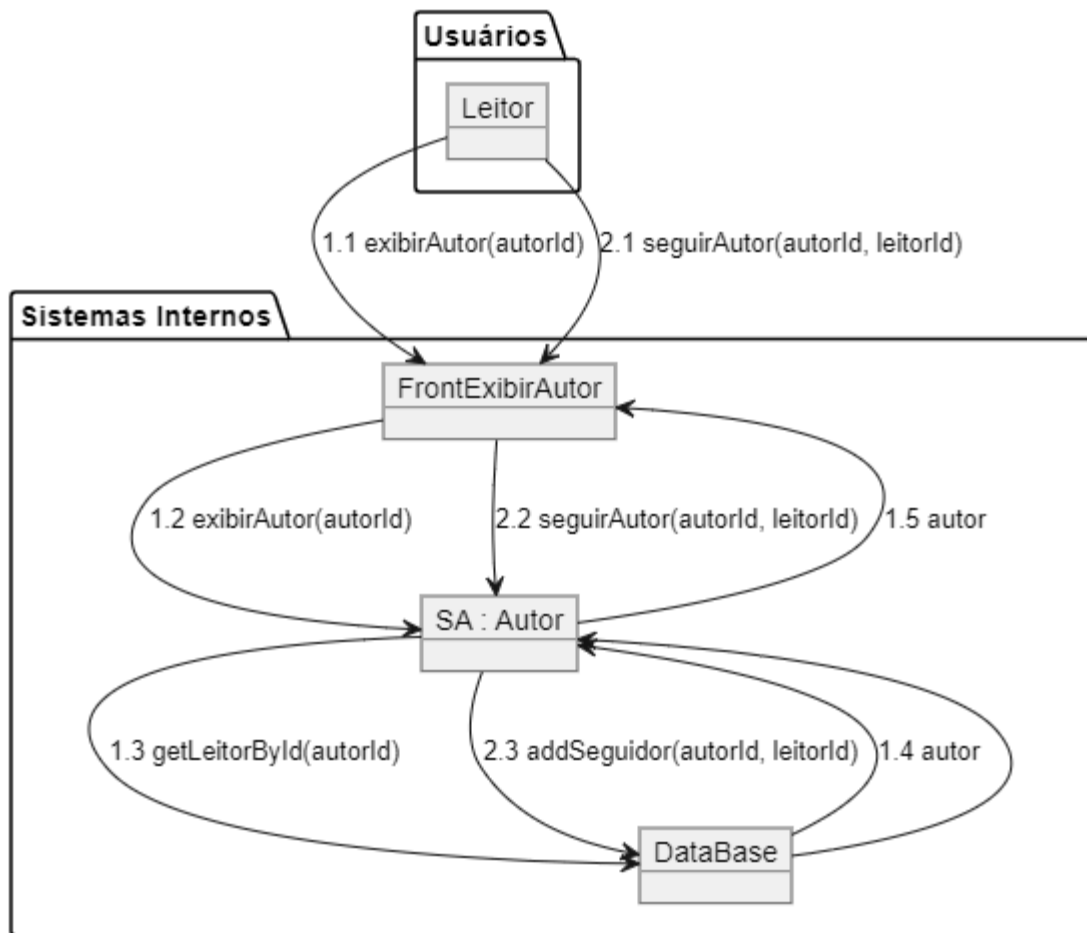
### 3.3 Diagrama de Classes



### 3.5.1 Diagramas de Comunicação - UC4 Avaliar Autor

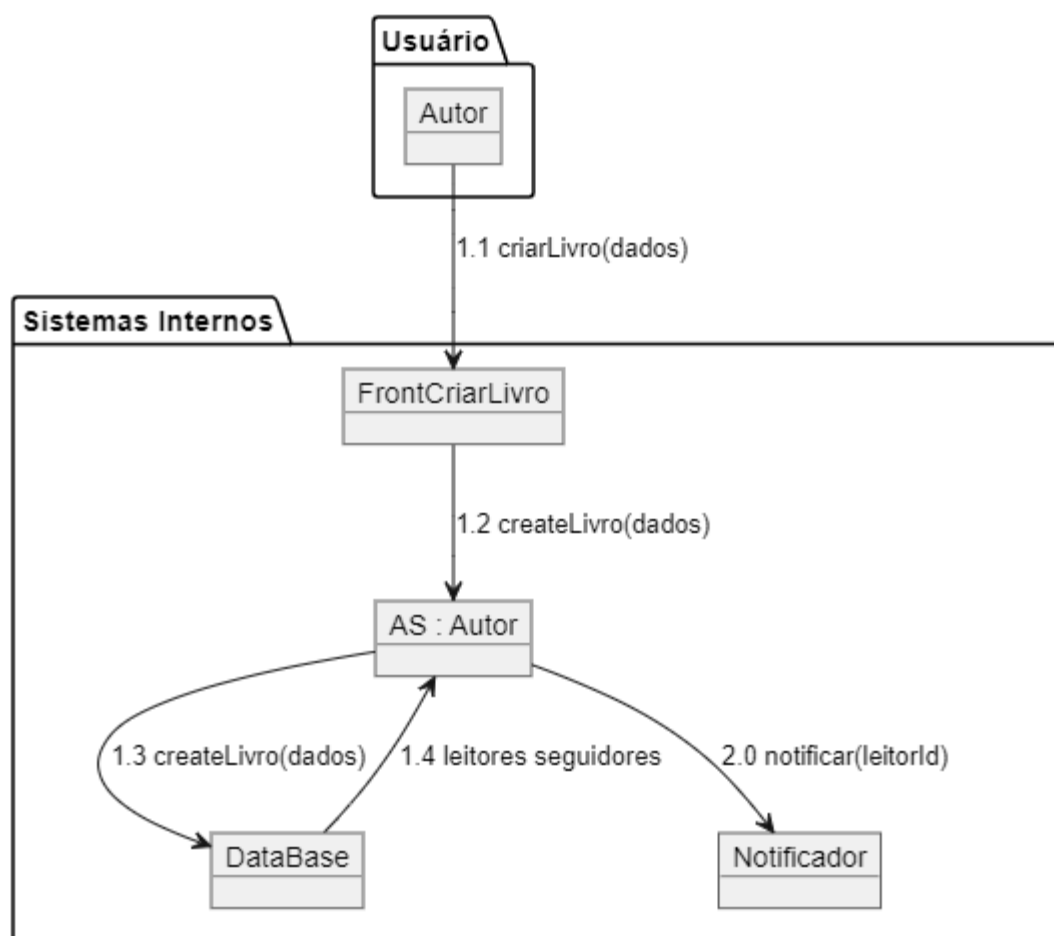


### 3.5.2 Diagrama de Comunicação - UC5 Seguir Autor

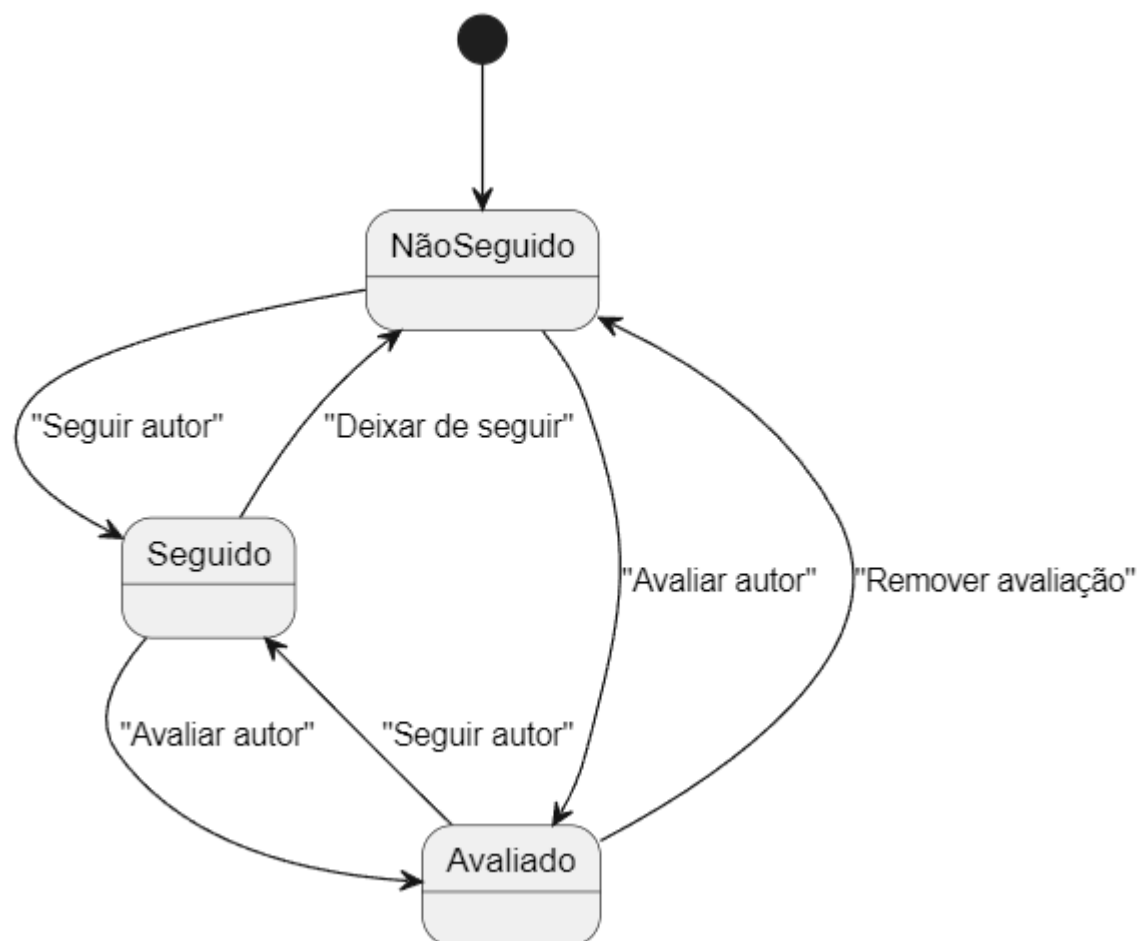




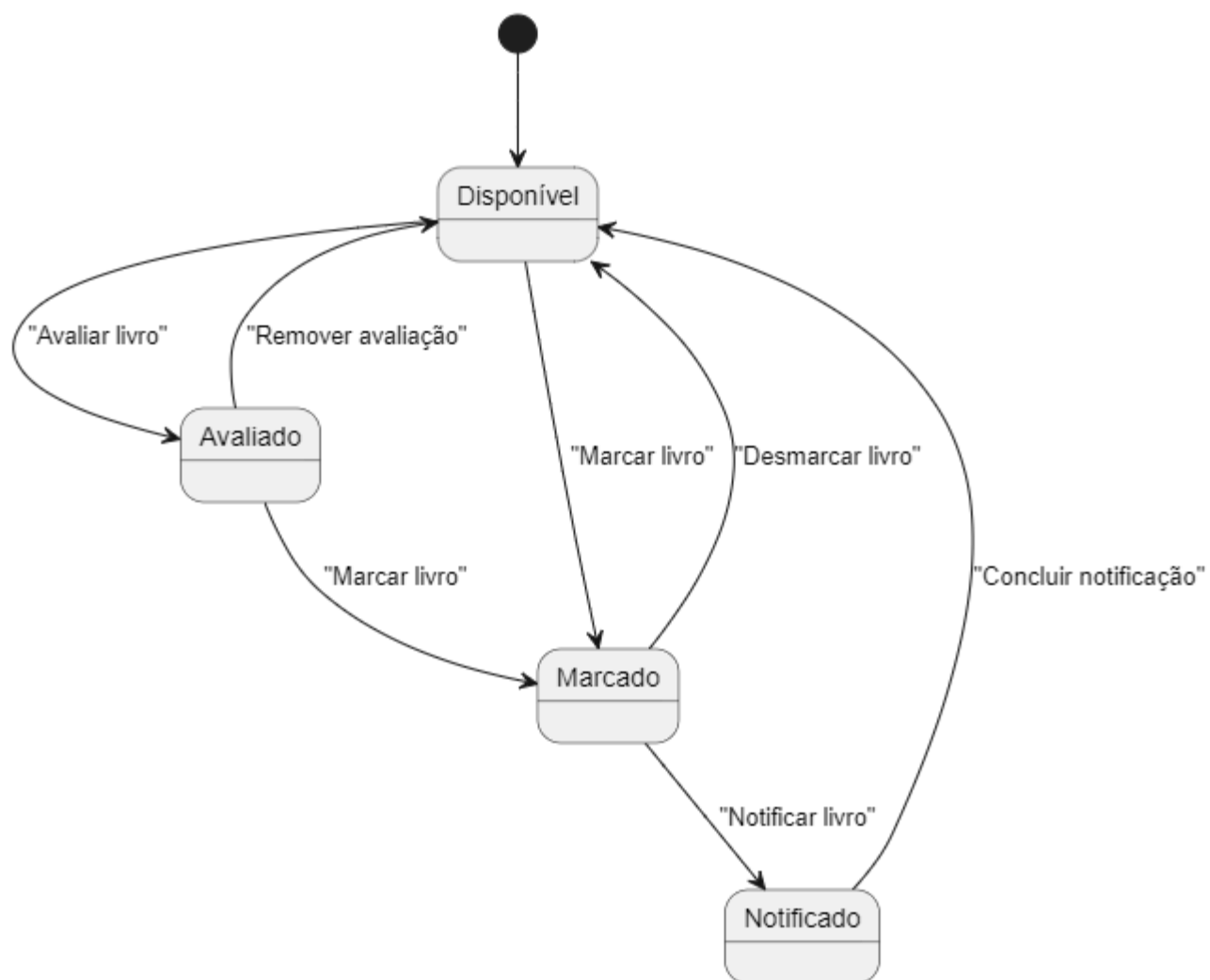
### 3.5.3 Diagrama de Comunicação - UC6 Notificar Livro



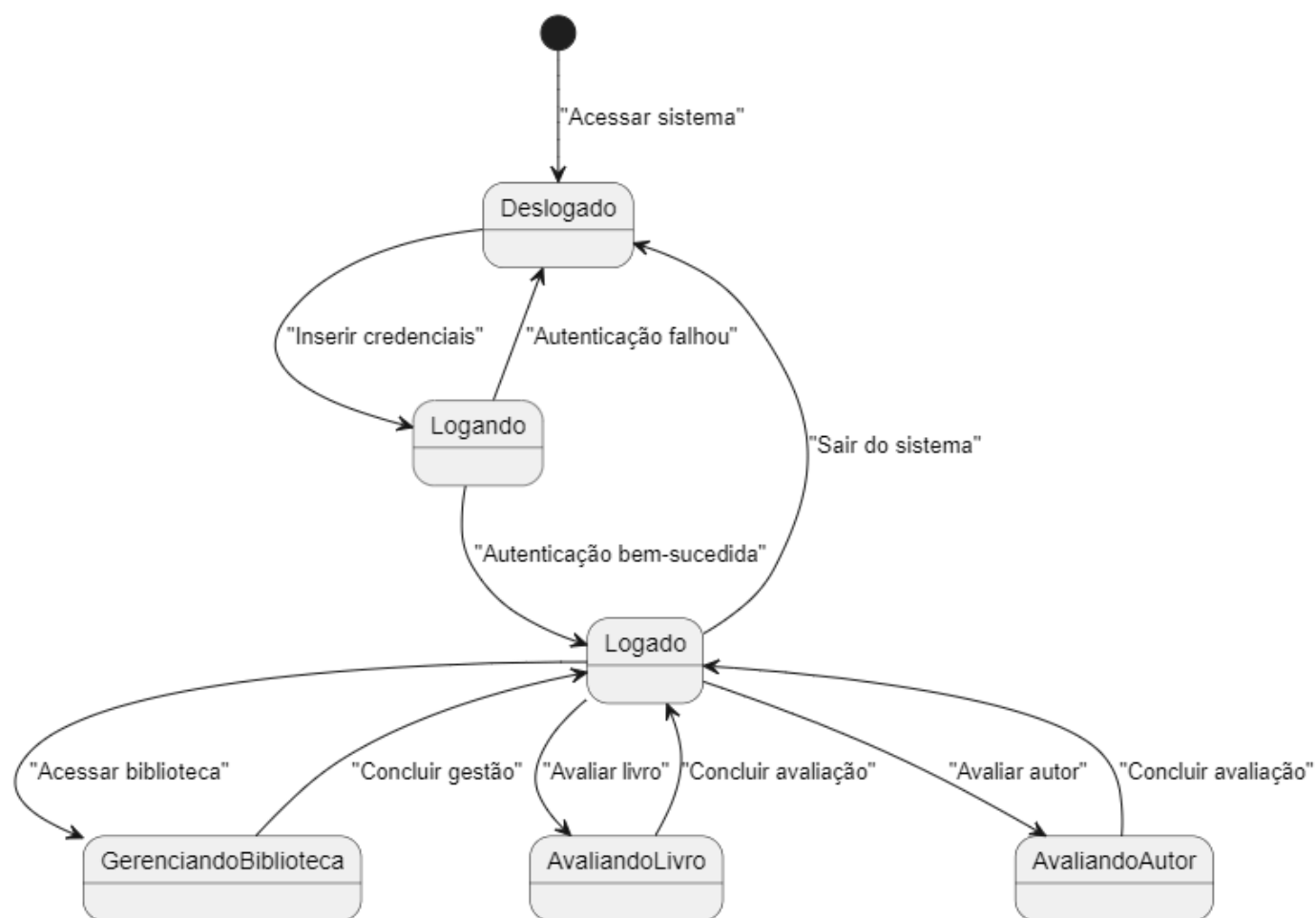
### 3.6.1 Diagrama de Estado - Autor



### 3.6.2 Diagrama de Estado - Livro



### 3.6.3 Diagrama de Estado - Usuário



## 4. Modelos de Dados

