
Documentação de Projeto

para o sistema

<HelpClass >

Versão 1.0

Projeto de sistema elaborado pelo aluno Guilherme de Almeida Santos
como parte da disciplina **Projeto de Software**.

<16/11/2024>

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 |
|---------|----------|------------------------------------|--------|
| Criação | 15/11/25 | Criação da documentação do projeto | V.01 |
| | | | |

1. Introdução

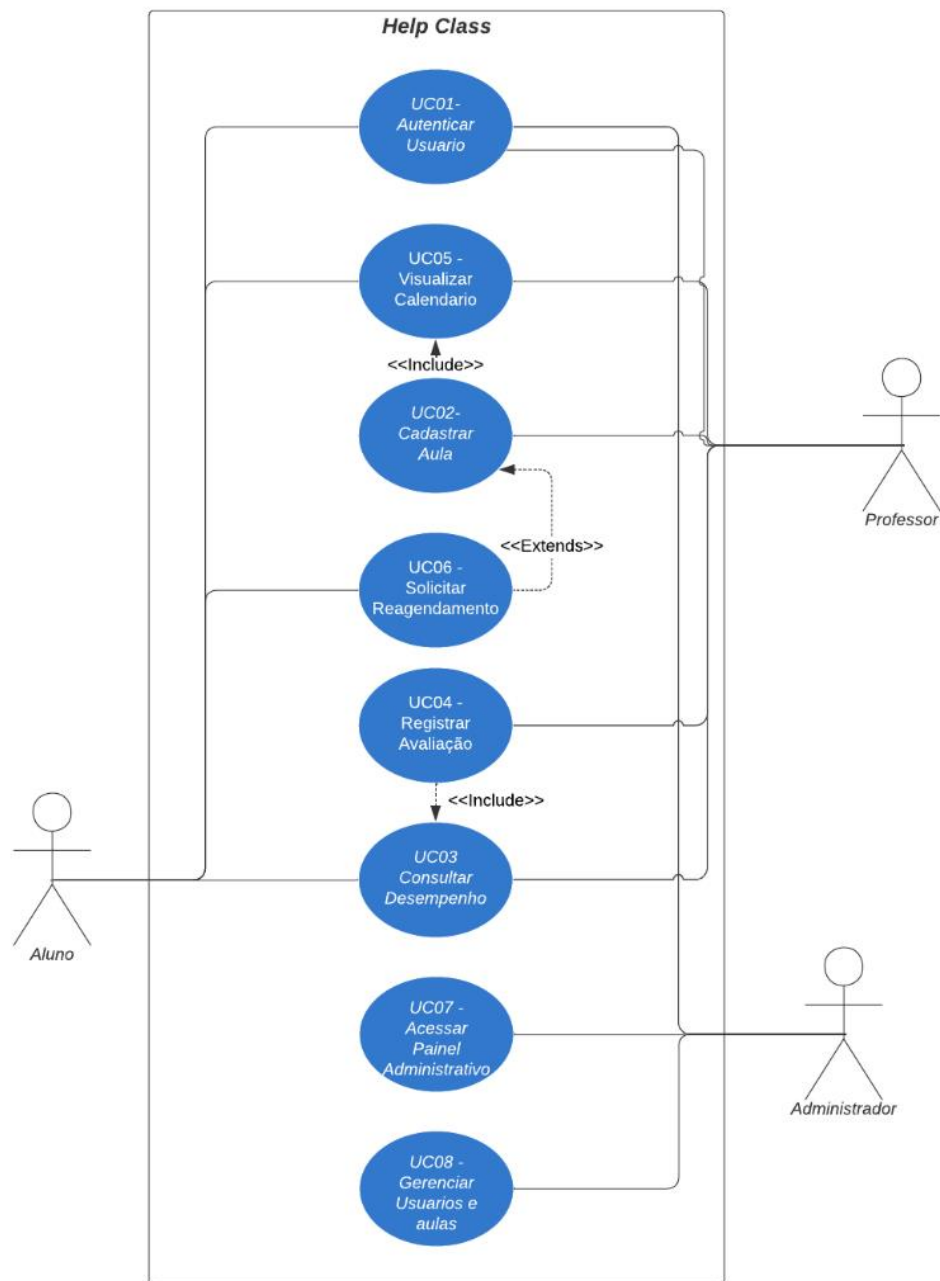
Este documento agrega: 1) a elaboração e revisão de modelos de domínio e 2) modelos de projeto para o sistema HelpClass. A referência principal para a descrição geral do problema, domínio e requisitos do sistema é o documento de especificação que descreve a visão de domínio do sistema.

2. Modelos de Usuário e Requisitos

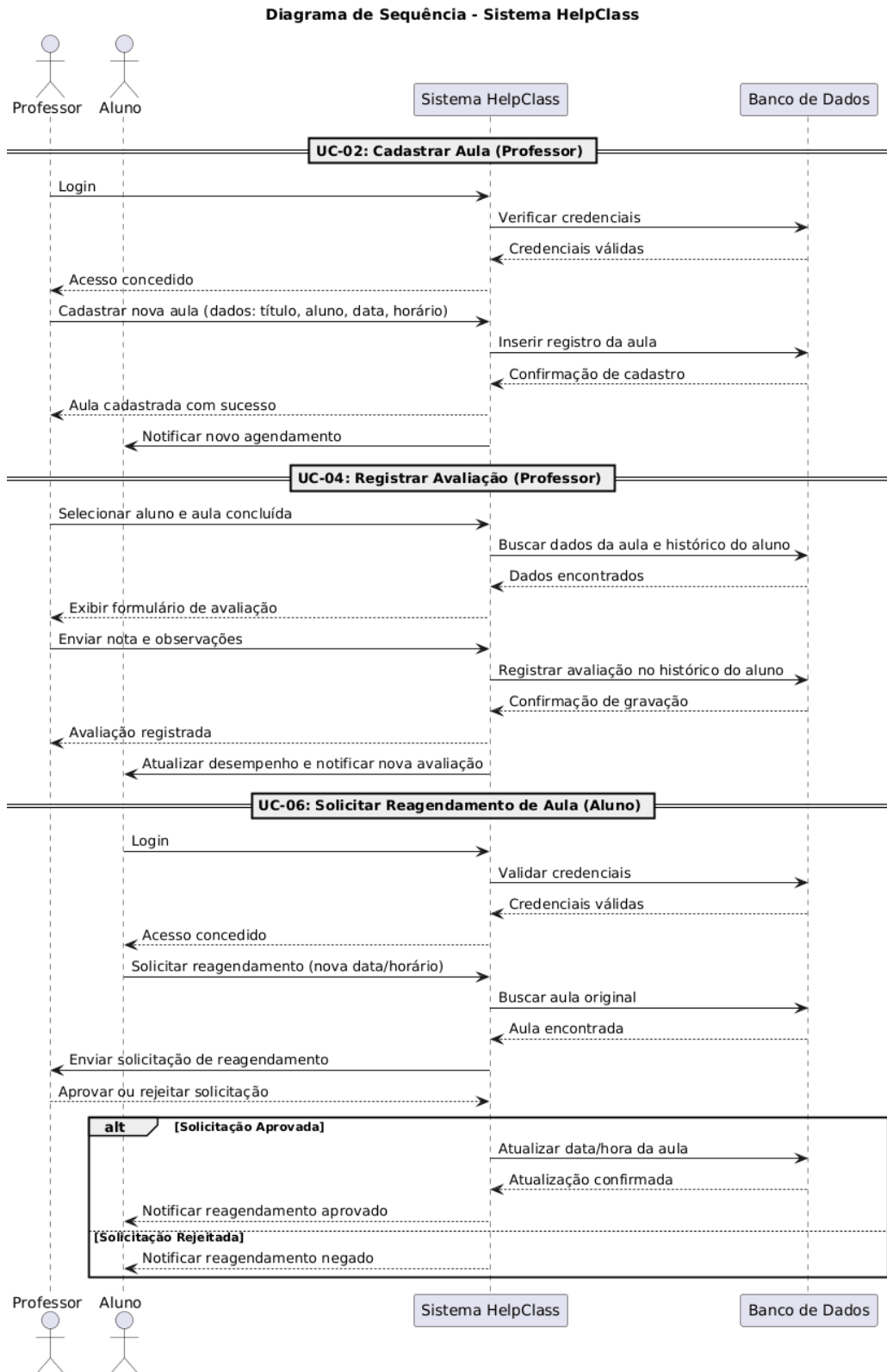
2.1 Descrição de Atores

O sistema de Gerenciamento de Aulas Particulares HelpClass é projetado para interações entre três atores principais. O Professor, que atua como docente, é responsável por gerenciar seus alunos, agendar aulas, configurar sua disponibilidade de horários e, crucialmente, avaliar e acompanhar o desempenho dos estudantes. O Aluno é o estudante que recebe o ensino, interagindo com o sistema para visualizar o calendário de aulas, solicitar reagendamentos e acompanhar sua própria evolução e indicadores de desempenho. Por fim, o Administrador possui a função de gerenciar todo o sistema, tendo acesso total e controle sobre a plataforma, normalmente por meio de uma interface administrativa como o Django Admin, garantindo a manutenção e o funcionamento adequado da aplicação.

2.2 Modelo de Casos de Uso



2.3 Diagrama de Sequência do Sistema



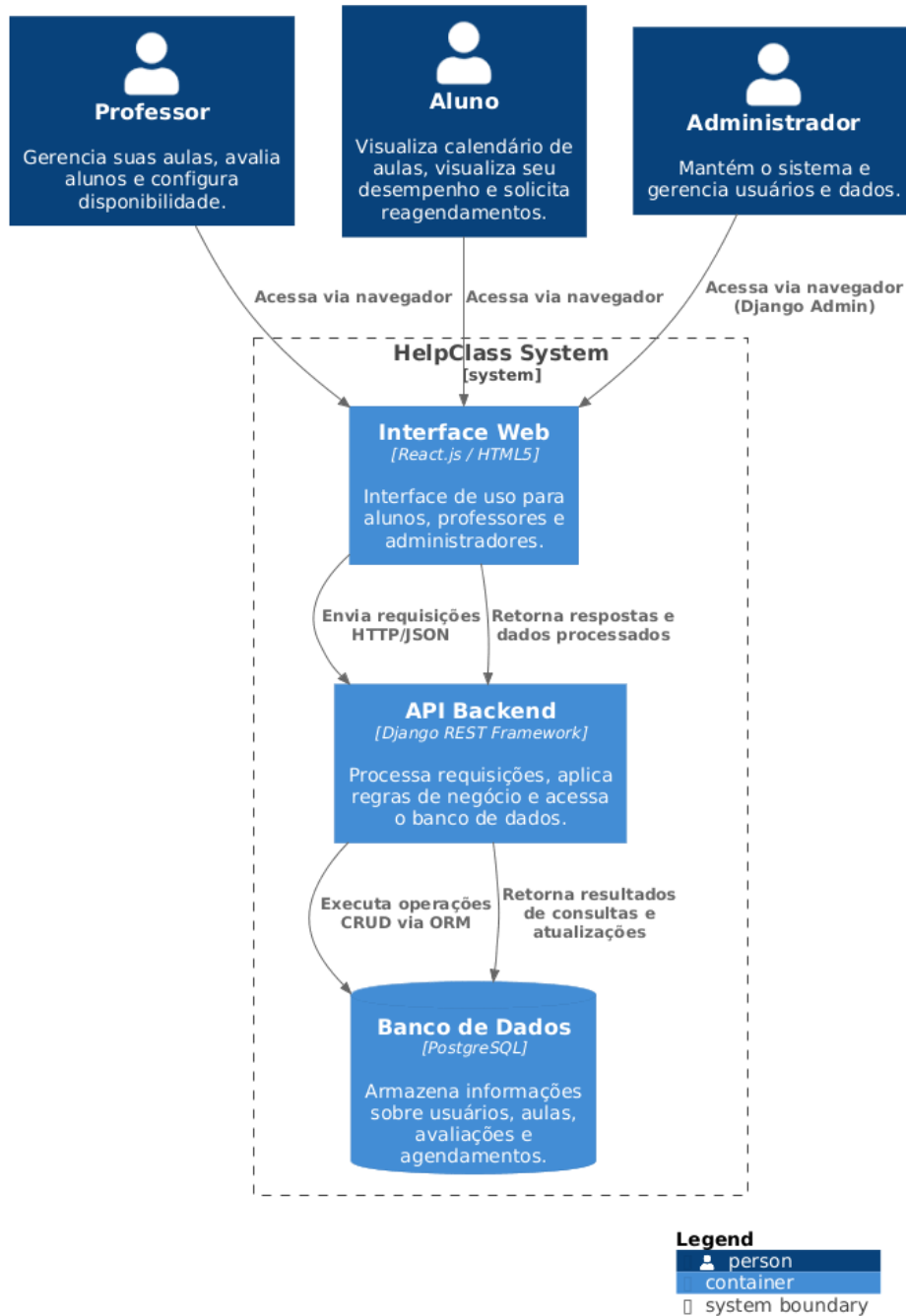
Formato para cada contrato de operação

| | |
|-----------------------------|--|
| Contrato | Cadastro de Aula |
| Operação | CadastrarAula() |
| Referências cruzadas | UC02 – Cadastrar Aula |
| Pré-condições | O Professor deve estar autenticado no sistema. Os dados obrigatórios da aula (aluno, data, hora, descrição) devem estar preenchidos. |
| Pós-condições | A nova aula é registrada no banco de dados. O Aluno é notificado sobre o novo agendamento. O calendário do Professor e do Aluno é atualizado. |
| | |
| Contrato | Registro de Avaliação |
| Operação | registrarAvaliacao() |
| Referências cruzadas | UC-04 – Registrar Avaliação |
| Pré-condições | O Professor deve estar autenticado. A aula deve estar concluída. O Aluno deve estar vinculado à aula. |
| Pós-condições | A avaliação é salva no histórico do aluno. O sistema atualiza os indicadores de desempenho. O aluno é notificado da nova avaliação registrada. |
| | |
| Contrato | Solicitação de Reagendamento |
| Operação | solicitarReagendamento() |
| Referências cruzadas | UC-06 – Solicitar Reagendamento de Aula |
| Pré-condições | O Aluno deve estar autenticado. A aula original deve estar previamente cadastrada. A nova data e horário devem estar disponíveis. |
| Pós-condições | O sistema registra a solicitação e notifica o Professor. Se aprovada, a data e o horário da aula são atualizados. O aluno recebe confirmação ou negação do pedido. |

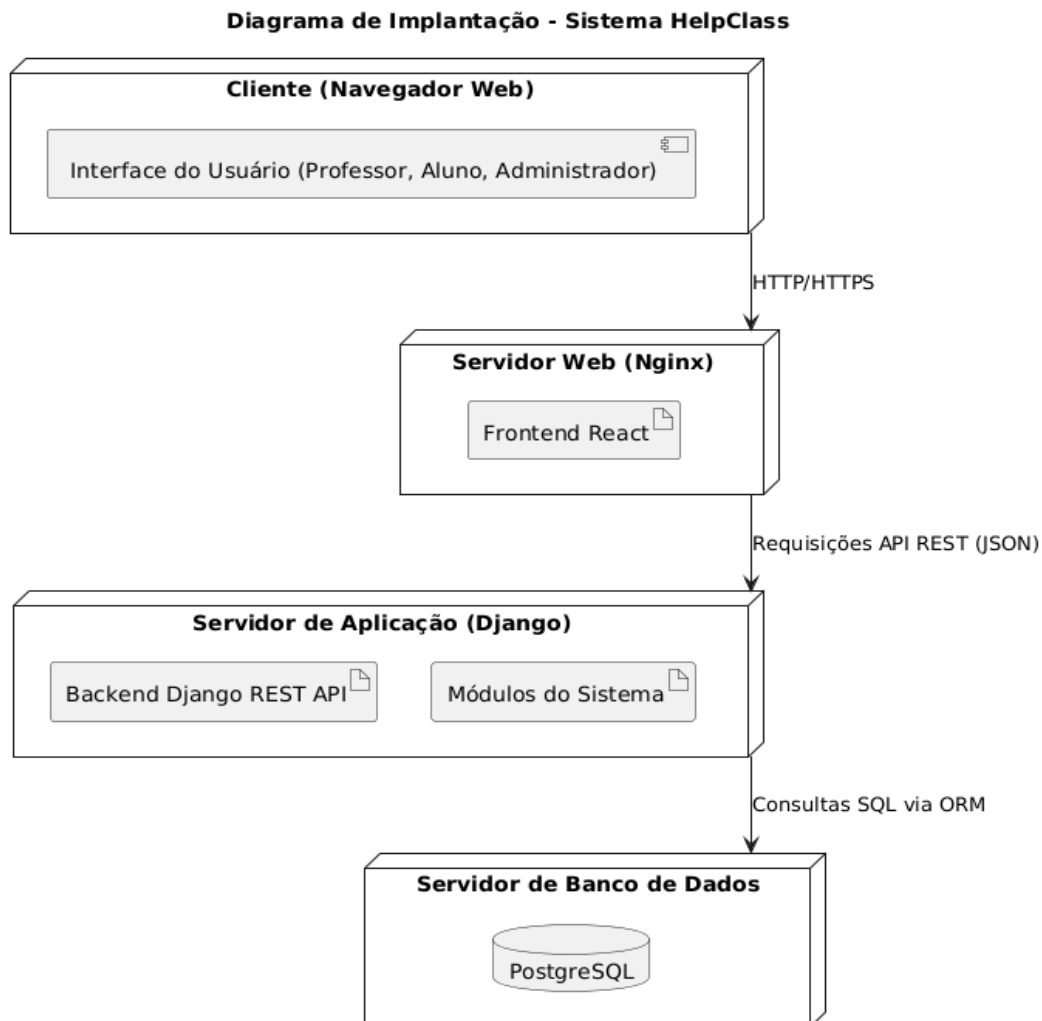
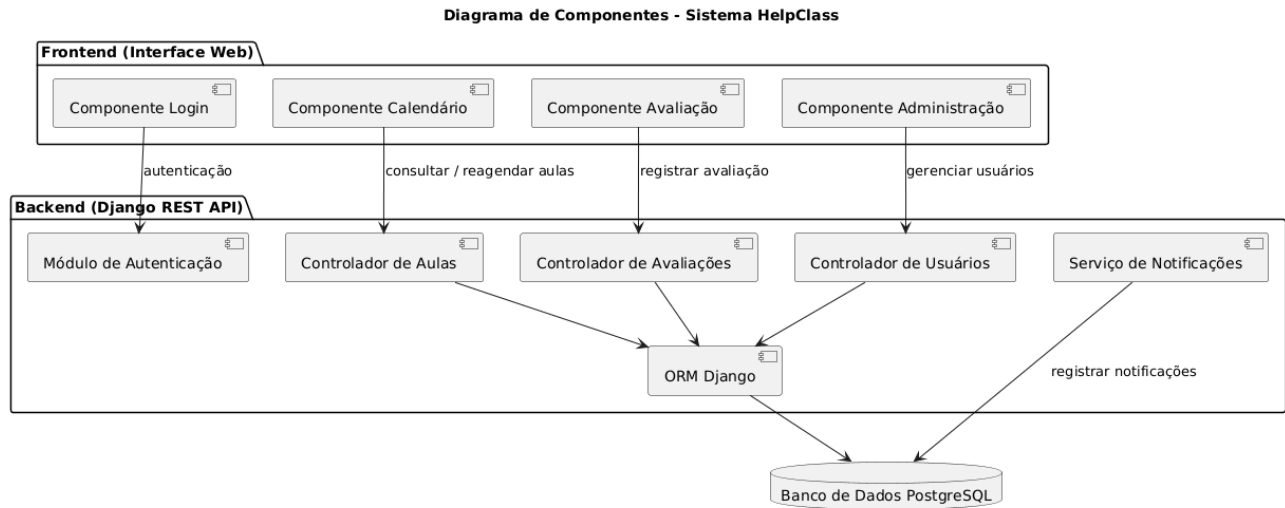
3. Modelos de Projeto

3.1 Arquitetura

3.1 Arquitetura - Sistema HelpClass (Níveis Contexto e Container)

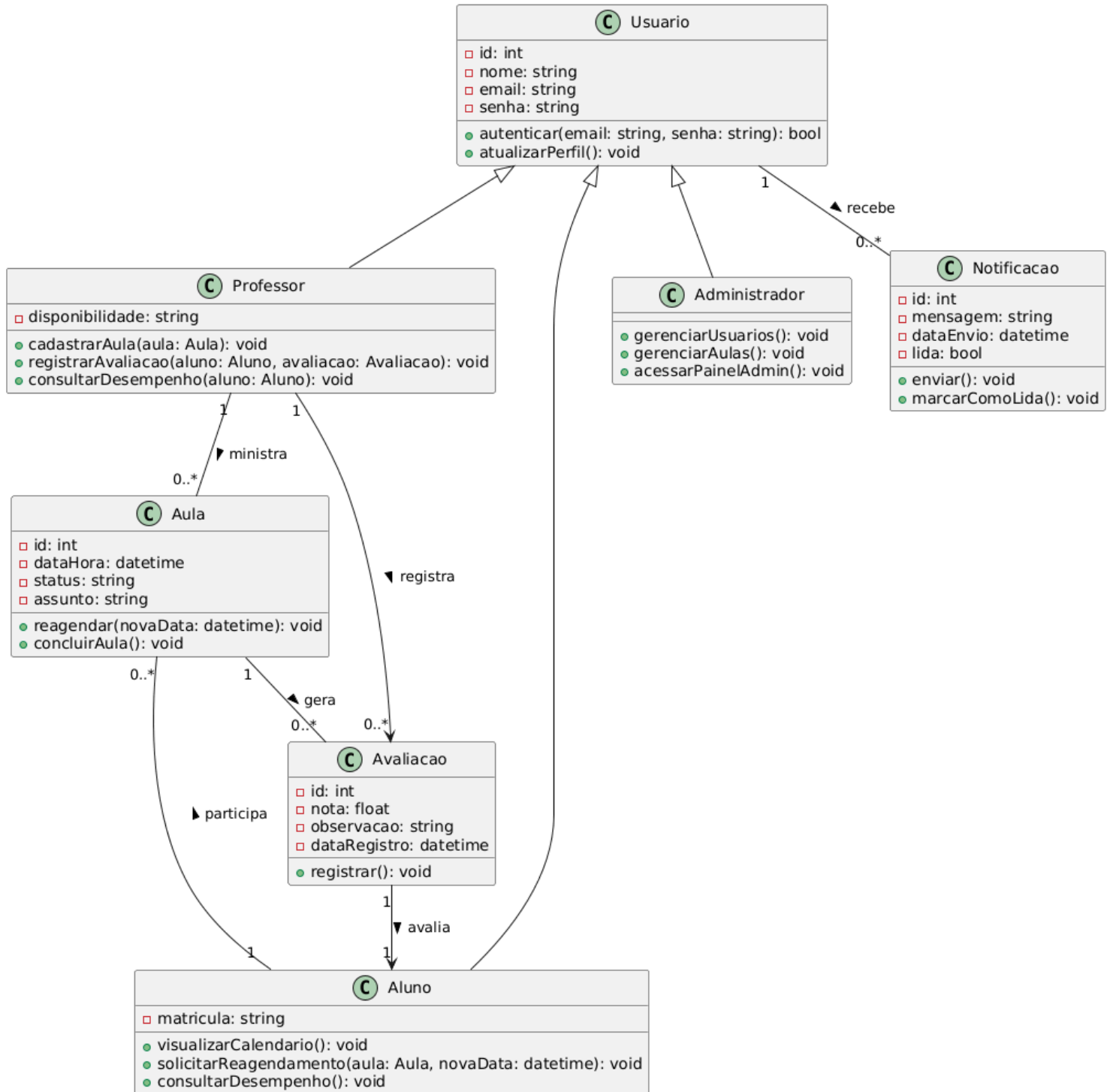


3.2 Diagrama de Componentes e Implantação.

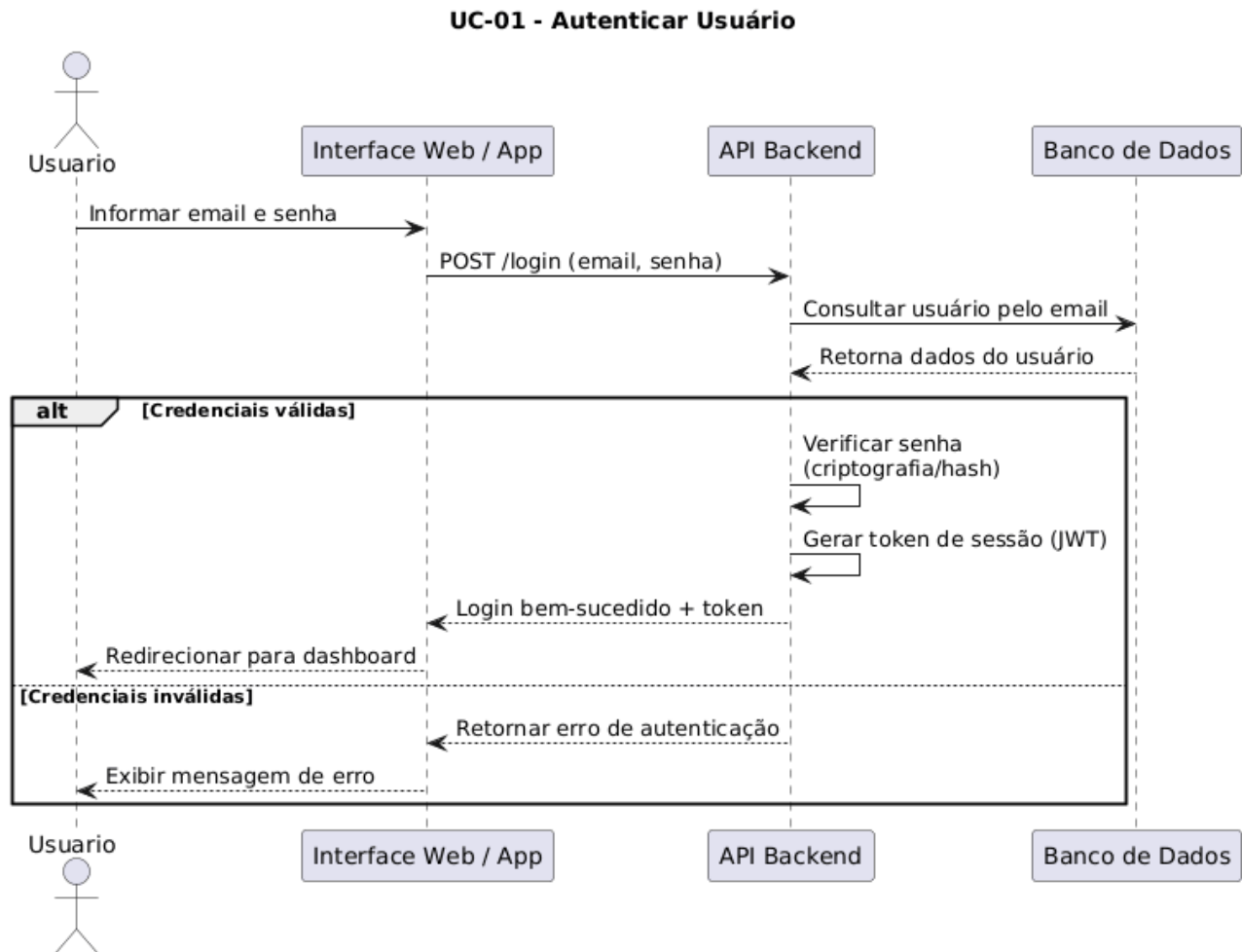


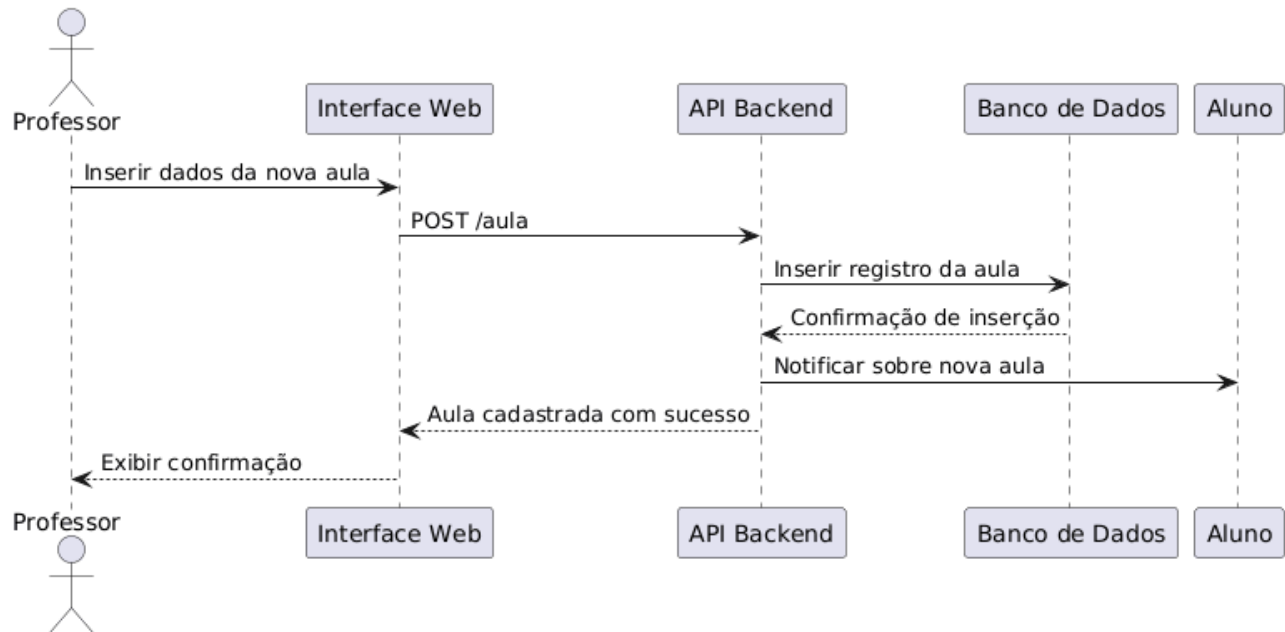
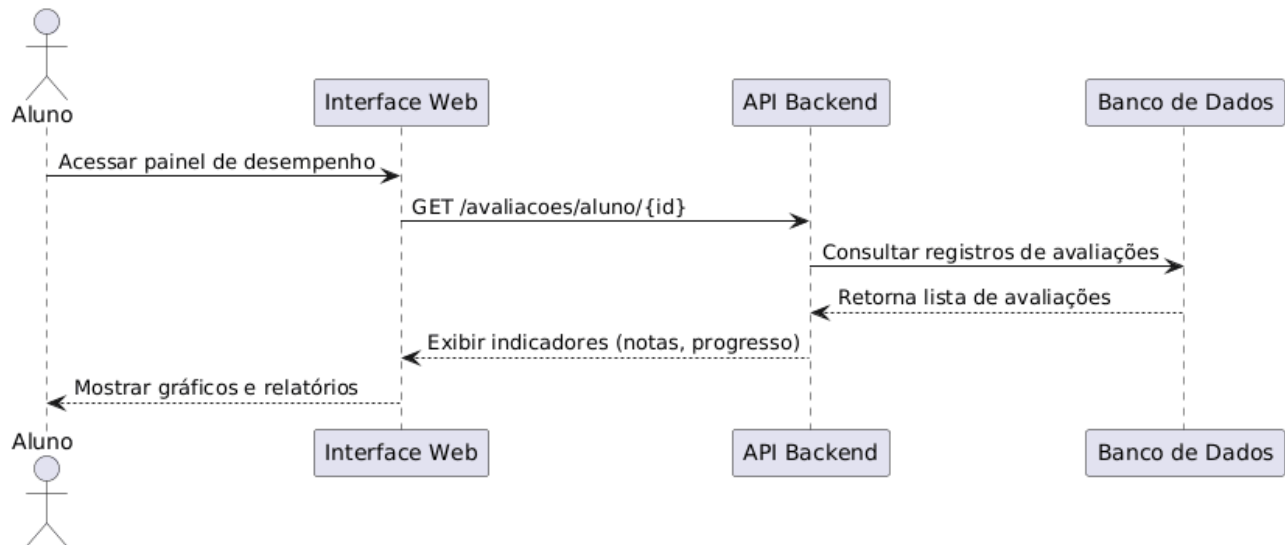
3.3 Diagrama de Classes

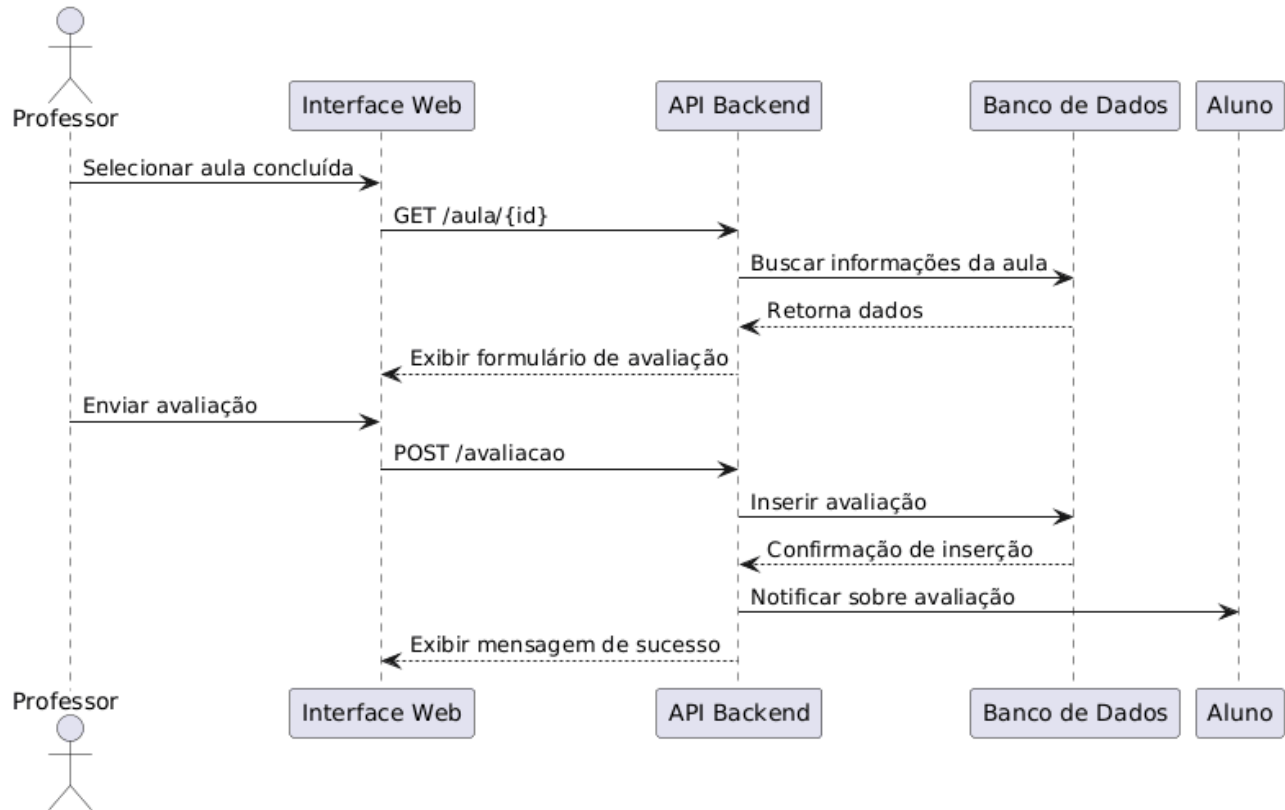
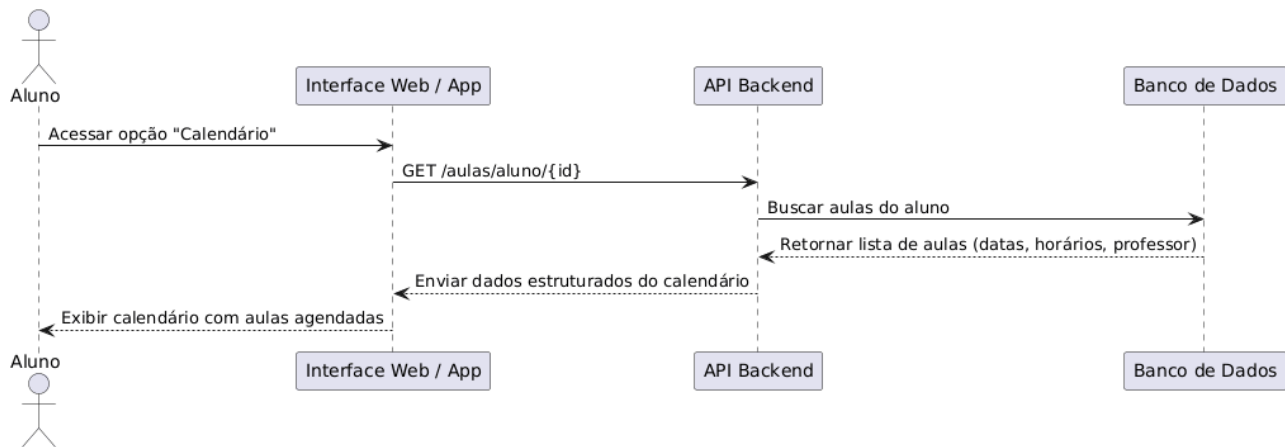
Diagrama de Classes - Sistema HelpClass



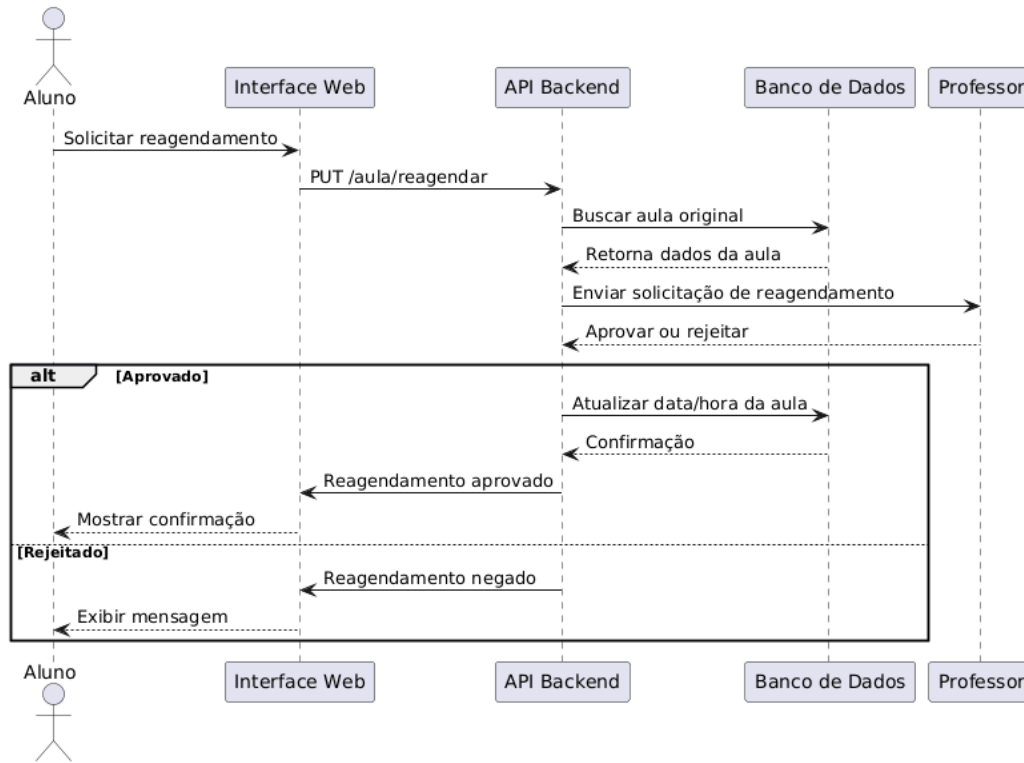
3.4 Diagramas de Sequência



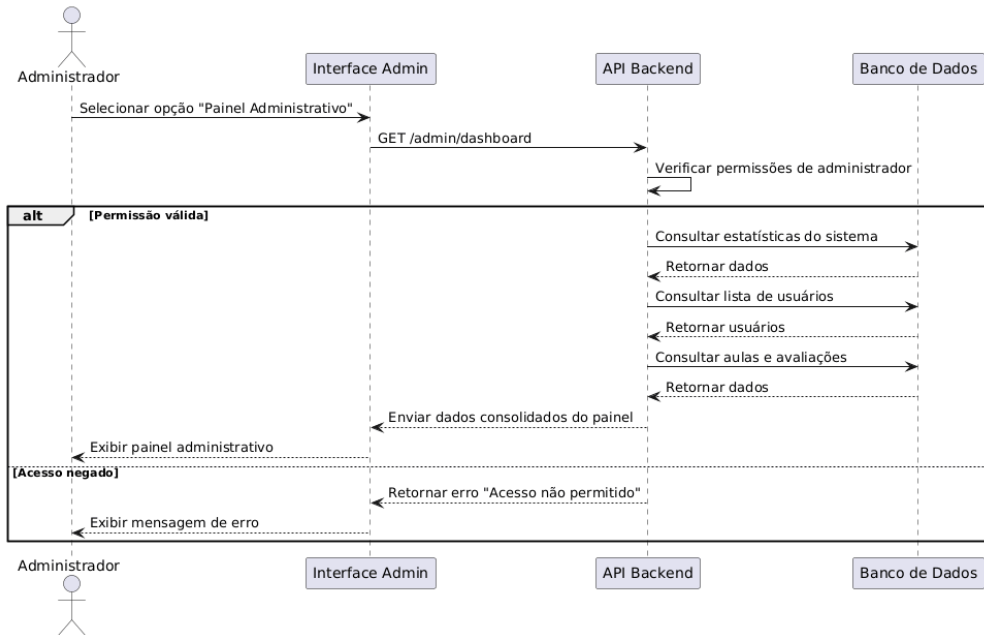
UC-02 - Cadastrar Aula**UC-03 - Consultar Desempenho**

UC-04 - Registrar Avaliação**UC-05 - Visualizar Calendário**

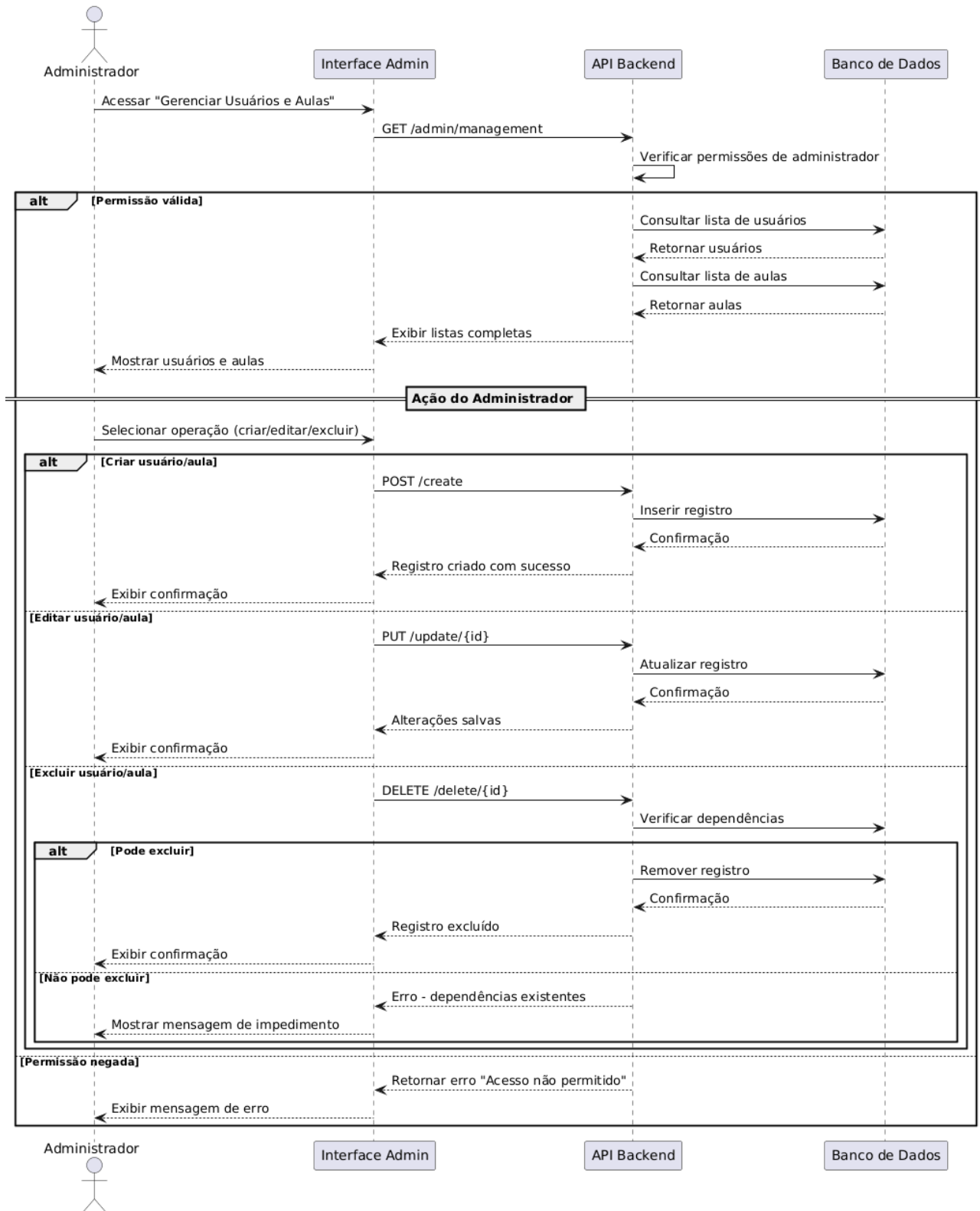
UC-06 - Solicitar Reagendamento



UC-07 - Acessar Paineil Administrativo

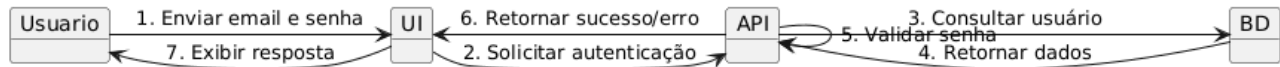


UC-08 - Gerenciar Usuários e Aulas

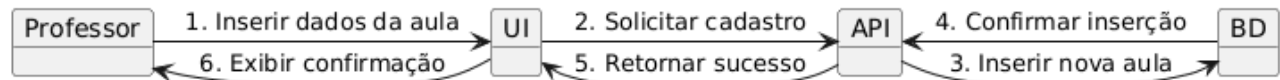


3.5 Diagramas de Comunicação

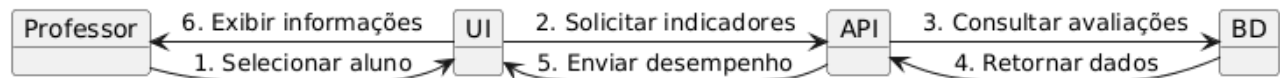
UC-01 - Comunicação: Autenticar Usuário



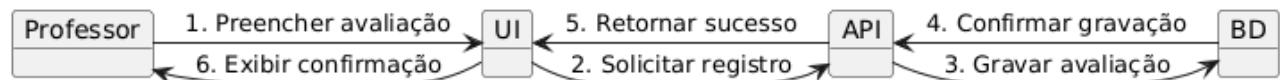
UC-02 - Comunicação: Cadastrar Aula



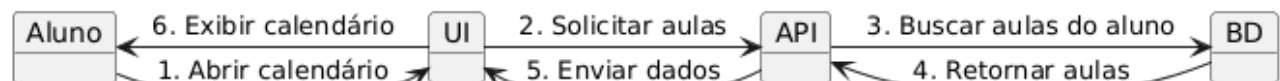
UC-03 - Comunicação: Consultar Desempenho



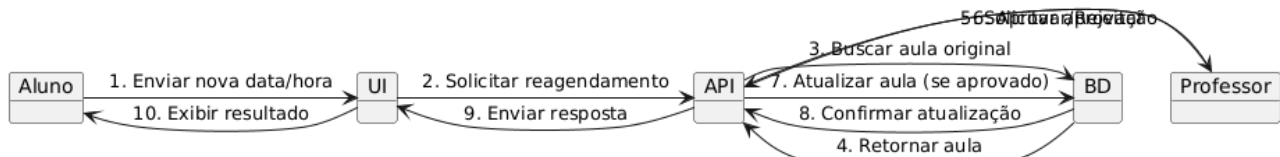
UC-04 - Comunicação: Registrar Avaliação



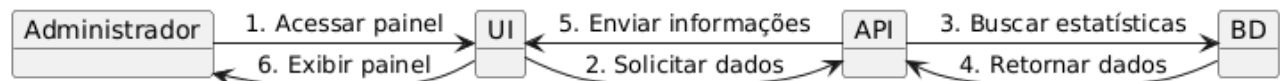
UC-05 - Comunicação: Visualizar Calendário



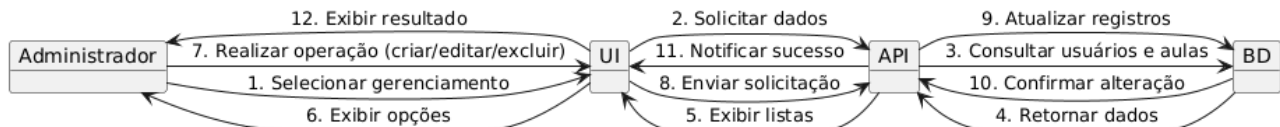
UC-06 - Comunicação: Solicitar Reagendamento



UC-07 - Comunicação: Acessar Painel Administrativo



UC-08 - Comunicação: Gerenciar Usuários e Aulas



3.6 Diagramas de Estados

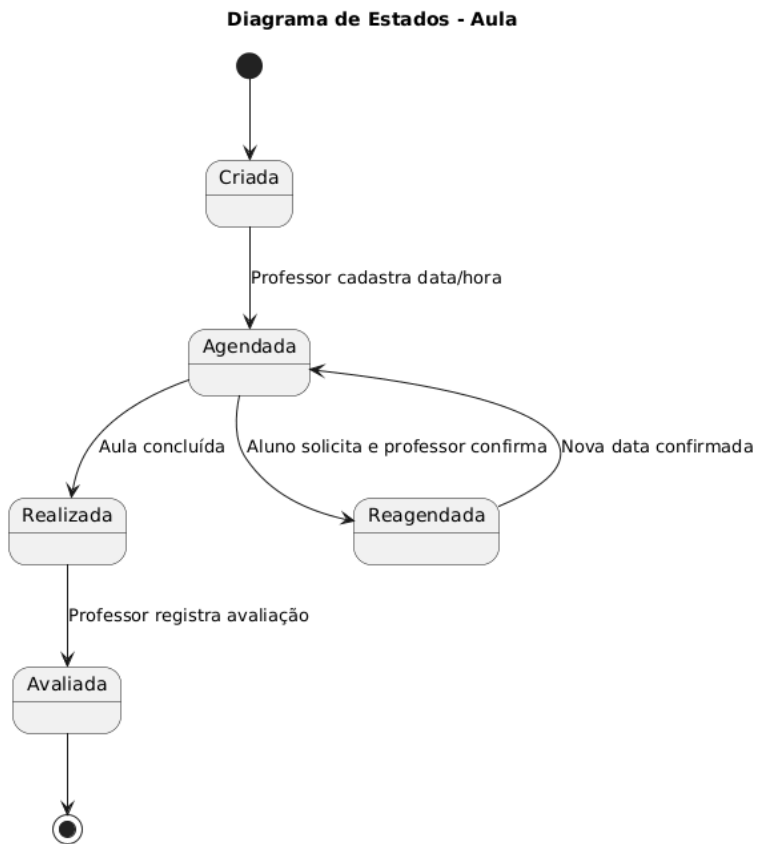
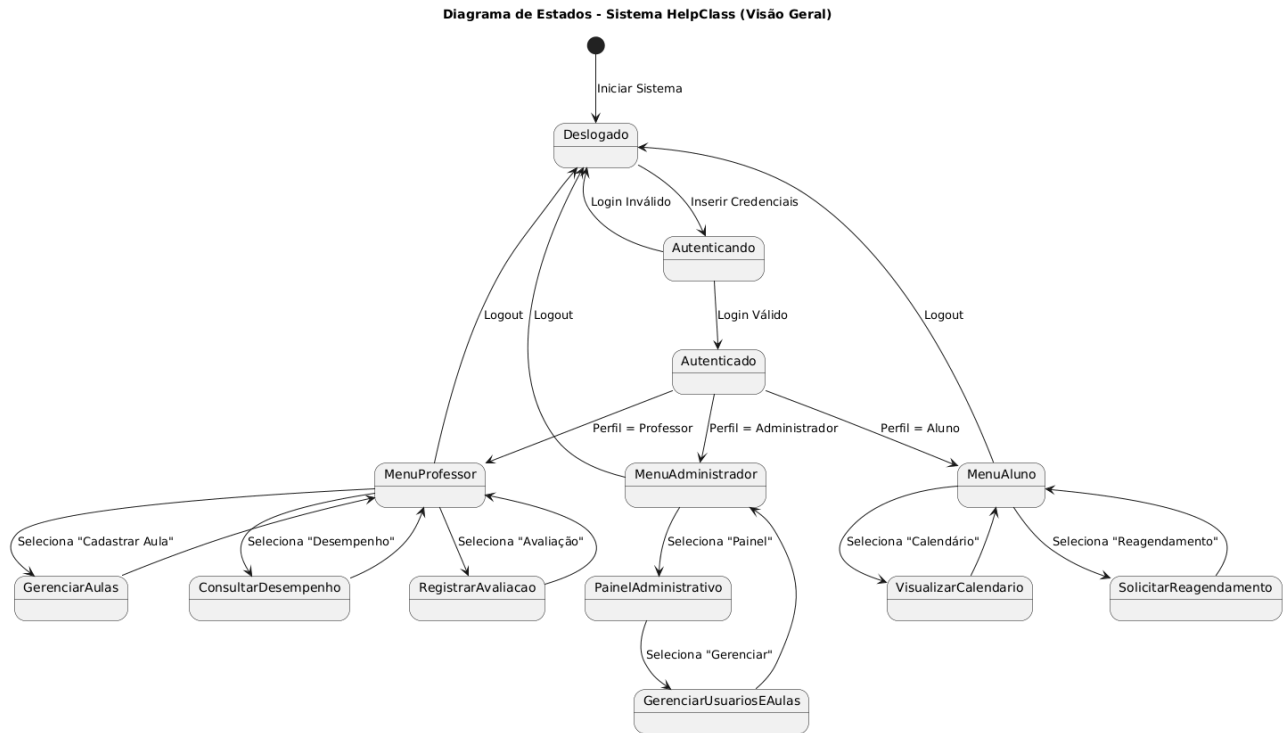
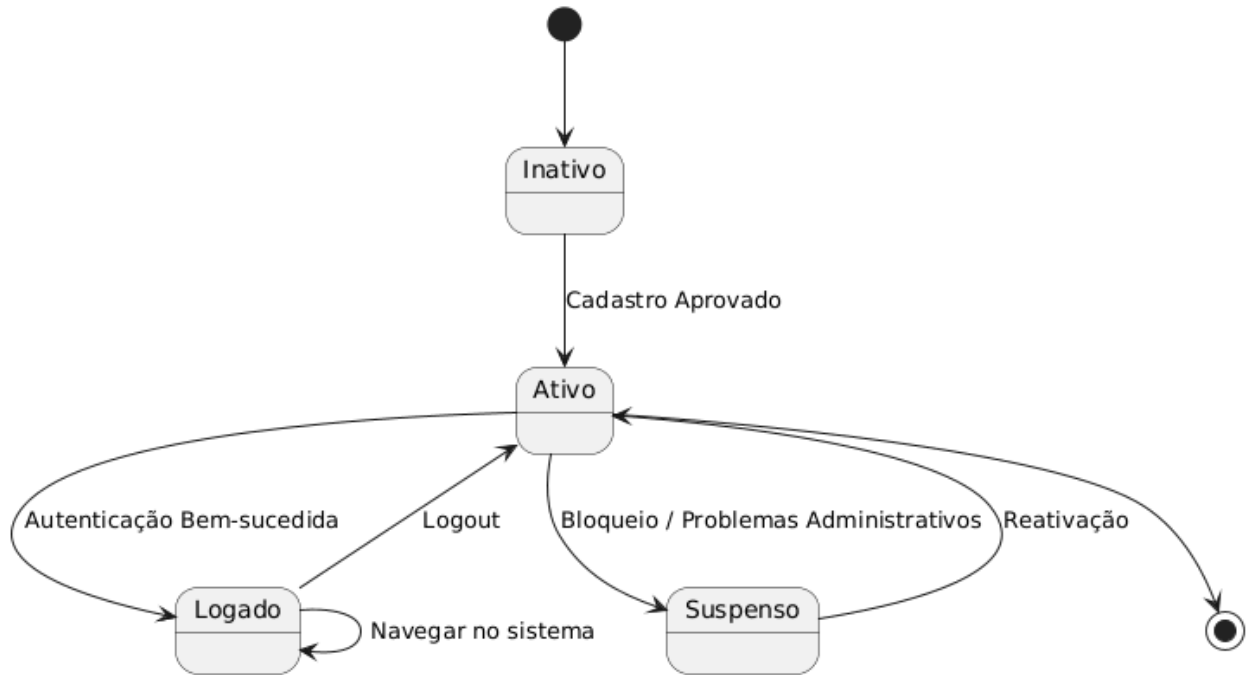
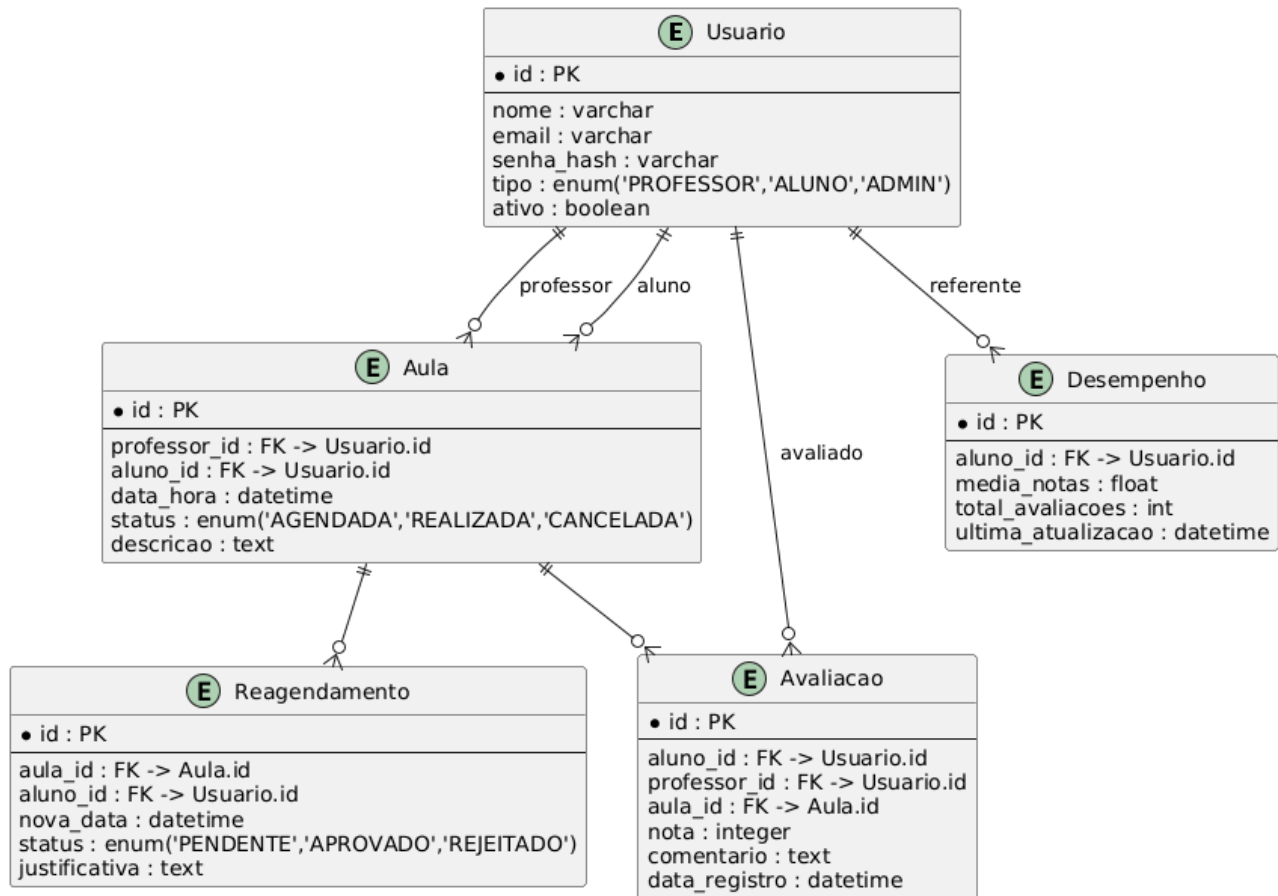


Diagrama de Estados - Usuário



4. Modelos de Dados



4.1 Estratégias de Mapeamento Objeto – Relacional (ORM)

4.1.1 Mapeamento 1–1 (One-to-One)

- Usuário (Aluno) → Desempenho
- Cada aluno possui um único registro de desempenho.

4.1.2 Mapeamento 1–N (One-to-Many)

- Professor → Aulas
- Aluno → Aulas
- Aula → Avaliações
- Aula → Reagendamentos

4.1.3 Benefícios do ORM no sistema

- Reduz dependência de SQL manual
- Evita inconsistências no banco
- Automatiza migrações
- Representa entidades como objetos
- Facilita testes e manutenção
- Permite versionamento estrutural