[**ÂNIMA - UNIVERSIDADE POTIGUAR**](https://analisederequisitos.com.br/)

[PROJETO](https://analisederequisitos.com.br/termo-de-abertura-do-projeto-exemplo) DE MODELAGEM DE SOFTWARE **SISTEMA DE CADASTRO DE ALUNOS**

**PREPARADO POR:**

Alexandre Lazoski Bastilho (Análise e desenvolvimento de sistemas)

Cinthia Raynara Ferreira da Silva (Análise e desenvolvimento de sistemas)

Felipe Vieira da Silva (Sistemas de Informação)

Ledson André de Lima e Silva (Análise e desenvolvimento de sistemas)

Marcelo Luiz Pinotti da Silva Junior (Análise e desenvolvimento de sistemas)

DOCUMENTO M-001

ÚLTIMA ATUALIZAÇÃO: 08/06/2025

[MODELO DE PROJETO DE MODELAGEM DE SOFTWARE](https://analisederequisitos.com.br/termo-de-abertura-do-projeto-exemplo)

ESTE MODELO FOI DISPONIBILIZADO PELO PROFESSOR JOSÉ PADILHA

Sumário

[1) Introdução 6](#_Toc200313253)

[2) Escopo do Produto 6](#_Toc200313254)

[3) Especificações do Hardware 7](#_Toc200313255)

[4) Abrangência e sistemas relacionados 7](#_Toc200313256)

[5) Descrição dos usuários 8](#_Toc200313257)

[5.1 Administrador 8](#_Toc200313258)

[5.2 Funcionário 8](#_Toc200313259)

[5.3 Professor 8](#_Toc200313260)

[5.4 Aluno 8](#_Toc200313261)

[6) Requisitos funcionais 9](#_Toc200313262)

[6.1 Gestão de Alunos 9](#_Toc200313263)

[6.2 Matrícula 9](#_Toc200313264)

[6.3 Gestão de Cursos 10](#_Toc200313265)

[6.4 Notas e Histórico 10](#_Toc200313266)

[6.5 Administração do Sistema 11](#_Toc200313267)

[6.6 Autenticação e Acesso 11](#_Toc200313268)

[7) Casos de Uso 12](#_Toc200313269)

[[USO001] Cadastrar Novo Aluno 12](#_Toc200313270)

[[USO002] Consultar Alunos 12](#_Toc200313271)

[[USO003] Atualizar Dados de Aluno 12](#_Toc200313272)

[[USO004] Excluir Aluno 12](#_Toc200313273)

[[USO005] Matricular Aluno em Curso 13](#_Toc200313274)

[[USO006] Atualizar Status de Matrícula 13](#_Toc200313275)

[[USO007] Cadastrar Novo Curso 13](#_Toc200313276)

[[USO008] Consultar Cursos 13](#_Toc200313277)

[[USO009] Atualizar Dados de Curso 14](#_Toc200313278)

[[USO010] Excluir Curso 14](#_Toc200313279)

[[USO011] Associar Professor a Curso 14](#_Toc200313280)

[[USO012] Registrar Nota 14](#_Toc200313281)

[[USO013] Atualizar Nota 15](#_Toc200313282)

[[USO014] Consultar Lista de Alunos em Seus Cursos 15](#_Toc200313283)

[[USO015] Consultar Notas de Alunos em Seus Cursos 15](#_Toc200313284)

[[USO016] Consultar Notas de Aluno 15](#_Toc200313285)

[[USO017] Consultar Próprias Notas 16](#_Toc200313286)

[[USO018] Cadastrar Novo Funcionário 16](#_Toc200313287)

[[USO019] Consultar Funcionários 16](#_Toc200313288)

[[USO020] Atualizar Dados de Funcionário 16](#_Toc200313289)

[[USO021] Excluir Funcionário 17](#_Toc200313290)

[[USO022] Autenticar-se no Sistema 17](#_Toc200313291)

[[USO023] Realizar Logout 17](#_Toc200313292)

[8) Requisitos não funcionais 17](#_Toc200313293)

[a) Usabilidade 17](#_Toc200313294)

[b) Desempenho 18](#_Toc200313295)

[c) Segurança 18](#_Toc200313296)

[d) Confiabilidade 19](#_Toc200313297)

[e) Manutenibilidade 19](#_Toc200313298)

[9) Artefatos UML 19](#_Toc200313299)

[Diagrama de Caso de Uso 21](#_Toc200313300)

[Diagrama de Classe 22](#_Toc200313301)

[Diagrama de Sequência 23](#_Toc200313302)

[Diagrama Entidade Relacionamento - DER 24](#_Toc200313303)

[10) Banco de Dados 24](#_Toc200313304)

[10.1 Relacionamento de tabelas 24](#_Toc200313305)

[Tabelas do Sistema 25](#_Toc200313306)

[Relacionamentos 25](#_Toc200313307)

[11) Conclusões 26](#_Toc200313308)

[12) Referências 26](#_Toc200313309)

# Introdução

Este documento de requisitos tem como objetivo apresentar o Sistema de Cadastro de Alunos aos envolvidos no projeto, incluindo desenvolvedores, gerentes e demais stakeholders. Ele fornece uma visão geral clara e concisa do sistema a ser desenvolvido, explicando seu propósito, os problemas que visa solucionar e o contexto de sua aplicação.

O sistema será utilizado por instituições de ensino para gerenciar, de forma eficiente e segura, os dados relacionados a alunos, cursos, matrículas, notas e usuários, garantindo controle de acesso conforme o perfil de cada tipo de usuário (Administrador, Funcionário, Professor e Aluno). As seções seguintes detalham os requisitos funcionais e não funcionais do sistema, além dos artefatos de modelagem necessários para a construção da solução proposta.

# Escopo do Produto

O Sistema de Cadastro de Alunos é uma solução desenvolvida para atender às necessidades de instituições de ensino no gerenciamento acadêmico. Seu principal propósito é digitalizar e organizar processos relacionados a alunos, cursos, matrículas, notas e usuários, garantindo agilidade, segurança e controle nas operações do cotidiano escolar.

O sistema permitirá o cadastro, atualização, consulta e exclusão de informações referentes a alunos, professores, cursos e funcionários, além do gerenciamento de matrículas e lançamento de notas. O acesso será segmentado por perfis de usuários (Administrador, Funcionário, Professor e Aluno), garantindo que cada usuário tenha acesso apenas às funcionalidades correspondentes ao seu papel.

Entre as funcionalidades principais, destacam-se:

* Gestão de usuários por perfil;
* Controle de matrículas e notas;
* Consulta de histórico acadêmico;
* Relatórios administrativos e acadêmicos.

O sistema **não contempla**:

* Integração com sistemas financeiros (como geração de boletos);
* Emissão de certificados digitais;
* Módulos de ensino a distância (EAD).

Aplicado no contexto institucional, o software visa melhorar o fluxo de informação, reduzir erros operacionais e otimizar o tempo da equipe pedagógica e administrativa. Ele está alinhado com os macroprocessos de gestão educacional e com as metas de transformação digital das instituições, promovendo maior controle, rastreabilidade e qualidade na gestão acadêmica.

# Especificações do Hardware

Para o correto funcionamento do Sistema de Cadastro de Alunos, recomenda-se o seguinte ambiente mínimo de hardware:

* **Processador:** Intel Core i3 ou equivalente (mínimo); Intel Core i5 ou superior (recomendado)
* **Memória RAM:** 4 GB (mínimo); 8 GB ou mais (recomendado para múltiplos usuários simultâneos)
* **Armazenamento:** 500 MB de espaço livre para instalação e dados do sistema
* **Disco Rígido:** SSD recomendado para melhor desempenho nas operações de leitura e escrita
* **Dispositivos de Entrada:** Teclado e mouse
* **Dispositivos de Saída:** Monitor com resolução mínima de 1280x720 pixels
* **Conectividade:** Acesso à internet para atualizações e backup em nuvem (se implementado)

O sistema foi desenvolvido em Java (versão 17 ou superior), sendo compatível com os principais sistemas operacionais, como Windows, Linux e macOS. O banco de dados utilizado é o MySQL, sendo necessário que o servidor ou máquina cliente possua o driver JDBC instalado e configurado corretamente.

Estas especificações visam garantir desempenho estável e compatível com o volume de dados e funcionalidades do sistema.

# Abrangência e sistemas relacionados

O Sistema de Cadastro de Alunos é **independente e totalmente autocontido**, não necessitando de comunicação com sistemas externos para desempenhar suas funções. Todo o processamento de dados, armazenamento e gerenciamento das informações ocorre dentro do próprio sistema, utilizando banco de dados local ou em servidor interno da instituição.

As principais funcionalidades do sistema incluem:

* Cadastro e gerenciamento de alunos, cursos, professores e funcionários;
* Controle de matrículas de alunos em cursos;
* Registro e consulta de notas por parte dos professores;
* Consulta de histórico acadêmico e dados pessoais pelos alunos;
* Controle de acesso baseado em perfis de usuário;
* Geração de relatórios acadêmicos.

Funcionalidades **não implementadas** nesta versão do sistema:

* Integração com sistemas de gestão financeira (como emissão de boletos e cobranças), por serem de responsabilidade de outras plataformas da instituição;
* Geração automatizada de certificados de conclusão, que poderá ser considerada em versões futuras;
* Integração com ambientes virtuais de aprendizagem (EAD), fora do escopo atual.

As funcionalidades citadas foram definidas com base nas necessidades imediatas da instituição e em seu fluxo de trabalho interno, priorizando autonomia, controle e facilidade de uso.

# Descrição dos usuários

O Sistema de Cadastro de Alunos foi projetado para atender diferentes perfis de usuários envolvidos na gestão acadêmica de uma instituição de ensino. Cada perfil possui responsabilidades específicas, e o sistema foi estruturado para facilitar suas atividades e resolver os principais desafios enfrentados no dia a dia, como a descentralização da informação, retrabalho manual, e falta de controle no acesso a dados sensíveis.

Os usuários do sistema são:

## **5.1 Administrador**

Responsável pela gestão de usuários do sistema, com foco principal na criação, atualização e inativação de contas de funcionários. Tem acesso total ao sistema, mas não atua diretamente sobre dados acadêmicos como alunos ou cursos.

**Desafios enfrentados:** controle e segurança na gestão de acessos e permissões.

## **5.2 Funcionário**

Usuário com papel fundamental na gestão operacional. É responsável por cadastrar, consultar, atualizar e excluir alunos, cursos e matrículas. Também pode gerar relatórios e consultar históricos acadêmicos.

**Desafios enfrentados:** lentidão em processos manuais, falta de padronização no cadastro e dificuldade para manter registros atualizados.

## **5.3 Professor**

Possui acesso às turmas e cursos que leciona. Pode consultar a lista de alunos matriculados, registrar e atualizar as notas. Não possui acesso a dados administrativos ou pessoais de outros usuários.

**Desafios enfrentados:** dificuldade de controle e lançamento de notas em tempo hábil e com segurança.

## **5.4 Aluno**

Usuário final que pode acessar o sistema para consultar suas notas, acompanhar o histórico acadêmico e atualizar seus próprios dados (exceto CPF). Possui acesso restrito, apenas às suas informações pessoais.

**Desafios enfrentados:** falta de transparência no acompanhamento do desempenho acadêmico e necessidade de atualização de dados sem depender da secretaria.

# Requisitos funcionais

A seguir são apresentados os requisitos funcionais do sistema, organizados por categorias conforme os perfis de usuários e funcionalidades correlacionadas. Cada requisito está identificado com um código único, e sua prioridade é classificada como Essencial, Importante ou Desejável, conforme a criticidade para o funcionamento do sistema.

## **6.1 Gestão de Alunos**

* **[RF001] – Cadastrar Aluno**  
  Permite que o funcionário cadastre um novo aluno no sistema, informando nome, CPF, e-mail, telefone e endereço.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Essencial
* **[RF002] – Consultar Alunos**  
  Permite listar todos os alunos cadastrados ou buscar por ID.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Essencial
* **[RF003] – Atualizar Aluno**  
  Permite editar os dados de um aluno existente.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Essencial
* **[RF004] – Excluir Aluno**  
  Permite remover um aluno do sistema.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Importante
* **[RF005] – Atualizar Dados Pessoais (Aluno)**  
  Permite que o aluno edite seus próprios dados (exceto CPF).  
  **Ator:** Aluno  
  **Prioridade:** ✅ Importante

## **6.2 Matrícula**

* **[RF006] – Realizar Matrícula**  
  Permite ao funcionário matricular um aluno em um curso.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Essencial
* **[RF007] – Atualizar Status da Matrícula**  
  Permite alterar o status da matrícula (ativo, trancado, concluído).  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Importante

## **6.3 Gestão de Cursos**

* **[RF008] – Cadastrar Curso**  
  Permite cadastrar um novo curso com nome, categoria e duração.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Essencial
* **[RF009] – Consultar Cursos**  
  Permite listar todos os cursos cadastrados.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Essencial
* **[RF010] – Excluir Curso**  
  Permite remover um curso do sistema.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Importante
* **[RF011] – Associar Professor ao Curso**  
  Permite vincular um professor a um curso específico.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Importante

## **6.4 Notas e Histórico**

* **[RF012] – Registrar Nota**  
  Permite que o professor lance a nota de um aluno.  
  **Ator:** Professor  
  **Prioridade:** ✅ Essencial
* **[RF013] – Atualizar Nota**  
  Permite ao professor editar uma nota previamente registrada.  
  **Ator:** Professor  
  **Prioridade:** ✅ Importante
* **[RF014] – Consultar Notas (Aluno)**  
  Permite que o aluno visualize suas próprias notas.  
  **Ator:** Aluno  
  **Prioridade:** ✅ Essencial
* **[RF015] – Consultar Histórico Acadêmico**  
  Permite ao aluno visualizar todas as suas notas e status de matrícula.  
  **Ator:** Aluno  
  **Prioridade:** ✅ Essencial
* **[RF016] – Consultar Histórico (Funcionário)**  
  Permite ao funcionário consultar o histórico de qualquer aluno.  
  **Ator:** Funcionário  
  **Prioridade:** ✅ Importante
* **[RF017] – Consultar Alunos por Curso**  
  Permite ao professor visualizar a lista de alunos matriculados em seus cursos.  
  **Ator:** Professor  
  **Prioridade:** ✅ Importante

## **6.5 Administração do Sistema**

* **[RF018] – Cadastrar Funcionário**  
  Permite ao administrador cadastrar novos funcionários.  
  **Ator:** Administrador  
  **Prioridade:** ✅ Essencial
* **[RF019] – Consultar Funcionários**  
  Permite listar todos os funcionários do sistema.  
  **Ator:** Administrador  
  **Prioridade:** ✅ Importante
* **[RF020] – Atualizar Funcionário**  
  Permite editar os dados de um funcionário.  
  **Ator:** Administrador  
  **Prioridade:** ✅ Importante
* **[RF021] – Excluir ou Inativar Funcionário**  
  Permite remover ou desativar um funcionário no sistema.  
  **Ator:** Administrador  
  **Prioridade:** ✅ Importante

## **6.6 Autenticação e Acesso**

* **[RF022] – Login de Usuário**  
  Permite autenticação de alunos, professores, funcionários e administradores.  
  **Ator:** Sistema  
  **Prioridade:** ✅ Essencial
* **[RF023] – Controle de Acesso**  
  Garante que cada perfil tenha acesso apenas às funcionalidades correspondentes.  
  **Ator:** Sistema  
  **Prioridade:** ✅ Essencial
* **[RF024] – Logout**  
  Permite ao usuário encerrar sua sessão de forma segura.  
  **Ator:** Sistema  
  **Prioridade:** ✅ Importante

# Casos de Uso

## **[USO001] Cadastrar Novo Aluno**

**Descrição:** Funcionário cadastra um novo aluno com dados obrigatórios.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Funcionário acessa “Cadastrar Aluno”;
2. Preenche nome, CPF, e-mail, telefone, endereço;
3. Sistema valida e salva os dados;
4. Mensagem de confirmação é exibida.

## **[USO002] Consultar Alunos**

**Descrição:** Funcionário visualiza alunos cadastrados, por lista ou busca.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Acessa “Consultar Alunos”;
2. Visualiza lista ou busca por nome/ID;
3. Sistema exibe dados do(s) aluno(s).

## **[USO003] Atualizar Dados de Aluno**

**Descrição:** Permite editar informações de um aluno já existente.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Seleciona aluno;
2. Edita dados desejados;
3. Confirma alteração;
4. Sistema atualiza o registro.

## **[USO004] Excluir Aluno**

**Descrição:** Remove o cadastro de um aluno.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Localiza aluno;
2. Solicita exclusão;
3. Sistema confirma e exclui.

## **[USO005] Matricular Aluno em Curso**

**Descrição:** Realiza matrícula de aluno em um curso ativo.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Escolhe aluno e curso;
2. Define status como “ativo”;
3. Sistema grava a matrícula.

## **[USO006] Atualizar Status de Matrícula**

**Descrição:** Altera o status da matrícula (trancado, concluído etc.).  
**Ator:** Funcionário  
**Fluxo principal:**

1. Acessa matrícula;
2. Altera status;
3. Sistema salva alteração.

## **[USO007] Cadastrar Novo Curso**

**Descrição:** Permite criar um novo curso no sistema.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Informa nome, categoria e duração;
2. Confirma criação;
3. Sistema valida e armazena.

## **[USO008] Consultar Cursos**

**Descrição:** Visualiza todos os cursos cadastrados.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Acessa opção “Consultar Cursos”;
2. Sistema exibe lista.

## **[USO009] Atualizar Dados de Curso**

**Descrição:** Edita dados de um curso já existente.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Seleciona curso;
2. Altera dados;
3. Confirma e salva.

## **[USO010] Excluir Curso**

**Descrição:** Remove permanentemente um curso.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Escolhe curso;
2. Solicita exclusão;
3. Sistema exclui curso.

## **[USO011] Associar Professor a Curso**

**Descrição:** Vincula um professor a um curso existente.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Seleciona curso;
2. Escolhe professor;
3. Confirma associação.

## **[USO012] Registrar Nota**

**Descrição:** Professor insere nota de aluno em disciplina.  
**Ator:** Professor  
**Fluxo principal:**

1. Escolhe curso e aluno;
2. Insere nota;
3. Sistema salva nota.

## **[USO013] Atualizar Nota**

**Descrição:** Permite editar nota já lançada.  
**Ator:** Professor  
**Fluxo principal:**

1. Localiza aluno/nota;
2. Altera valor;
3. Confirma e salva.

## **[USO014] Consultar Lista de Alunos em Seus Cursos**

**Descrição:** Visualiza todos os alunos matriculados em turmas do professor.  
**Ator:** Professor  
**Fluxo principal:**

1. Acessa “Minhas Turmas”;
2. Visualiza alunos por curso.

## **[USO015] Consultar Notas de Alunos em Seus Cursos**

**Descrição:** Visualiza notas lançadas nos cursos em que leciona.  
**Ator:** Professor  
**Fluxo principal:**

1. Escolhe curso;
2. Visualiza lista de notas.

## **[USO016] Consultar Notas de Aluno**

**Descrição:** Funcionário acessa o histórico de notas de um aluno.  
**Ator:** Funcionário  
**Fluxo principal:**

1. Busca aluno;
2. Exibe notas de todas as disciplinas.

## **[USO017] Consultar Próprias Notas**

**Descrição:** Aluno visualiza suas notas e desempenho.  
**Ator:** Aluno  
**Fluxo principal:**

1. Faz login;
2. Acessa “Minhas Notas”;
3. Sistema exibe dados.

## **[USO018] Cadastrar Novo Funcionário**

**Descrição:** Cria um novo cadastro de funcionário.  
**Ator:** Administrador  
**Fluxo principal:**

1. Preenche dados do novo funcionário;
2. Salva cadastro.

## **[USO019] Consultar Funcionários**

**Descrição:** Lista todos os funcionários ativos no sistema.  
**Ator:** Administrador  
**Fluxo principal:**

1. Acessa “Funcionários”;
2. Visualiza lista.

## **[USO020] Atualizar Dados de Funcionário**

**Descrição:** Permite editar dados de um funcionário.  
**Ator:** Administrador  
**Fluxo principal:**

1. Seleciona funcionário;
2. Altera dados;
3. Salva registro.

## **[USO021] Excluir Funcionário**

**Descrição:** Remove ou inativa um funcionário do sistema.  
**Ator:** Administrador  
**Fluxo principal:**

1. Escolhe funcionário;
2. Executa ação de exclusão.

## **[USO022] Autenticar-se no Sistema**

**Descrição:** Realiza login com credenciais válidas.  
**Ator:** Aluno, Professor, Funcionário ou Administrador  
**Fluxo principal:**

1. Usuário informa login e senha;
2. Sistema valida e direciona conforme perfil.

## **[USO023] Realizar Logout**

**Descrição:** Finaliza a sessão ativa do usuário.  
**Ator:** Todos os perfis  
**Fluxo principal:**

1. Clica em “Sair”;
2. Sessão é encerrada com segurança.

# Requisitos não funcionais

## **a) Usabilidade**

**[NF001] Interface Intuitiva**  
A interface do sistema deve ser intuitiva e de fácil aprendizado para os diferentes perfis de usuários.  
**Prioridade:** ✔ Essencial

**[NF002] Feedback ao Usuário**  
O sistema deve fornecer feedback claro ao usuário após a realização de ações (mensagens de sucesso, erro, confirmação).  
**Prioridade:** ✔ Essencial

**[NF003] Tempo de Resposta**  
O tempo de resposta para operações comuns (consultas, envio de formulários) não deve exceder 3 segundos em condições normais.  
**Prioridade:** ✔ Importante

**[NF004] Compatibilidade com Navegadores**  
O sistema deve funcionar nos navegadores web modernos (últimas duas versões de Chrome, Firefox, Edge e Safari).  
**Prioridade:** ✔ Desejável

## **b) Desempenho**

**[NF005] Usuários Simultâneos**  
O sistema deve suportar múltiplos usuários realizando operações simultâneas sem degradação significativa de desempenho.  
**Prioridade:** ✔ Essencial

**[NF006] Consultas Rápidas**  
Listagens de alunos ou cursos devem retornar resultados em até 5 segundos.  
**Prioridade:** ✔ Importante

## **c) Segurança**

**[NF007] Armazenamento Seguro de Dados**  
Senhas e dados sensíveis devem ser armazenados de forma segura (ex: com hash criptográfico).  
**Prioridade:** ✔ Essencial

**[NF008] Proteção contra Ataques**  
O sistema deve implementar proteções contra SQL Injection, XSS e outros ataques comuns.  
**Prioridade:** ✔ Essencial

**[NF009] Controle de Acesso**  
Acesso às funcionalidades será controlado por autenticação e autorização por perfil.  
**Prioridade:** ✔ Essencial

**[NF010] Registro de Logs**  
Ações sensíveis e acessos ao sistema devem ser registrados para auditoria.  
**Prioridade:** ✔ Importante

## **d) Confiabilidade**

**[NF011] Disponibilidade do Sistema**  
O sistema deve estar disponível para uso 24h por dia.  
**Prioridade:** ✔ Importante

**[NF012] Backup Regular dos Dados**  
O sistema deve realizar backups periódicos dos dados para evitar perda de informações.  
**Prioridade:** ✔ Importante

**[NF013] Validação de Dados**  
Mensagens de erro devem ser claras ao informar dados incorretos, permitindo correção fácil pelo usuário.  
**Prioridade:** ✔ Essencial

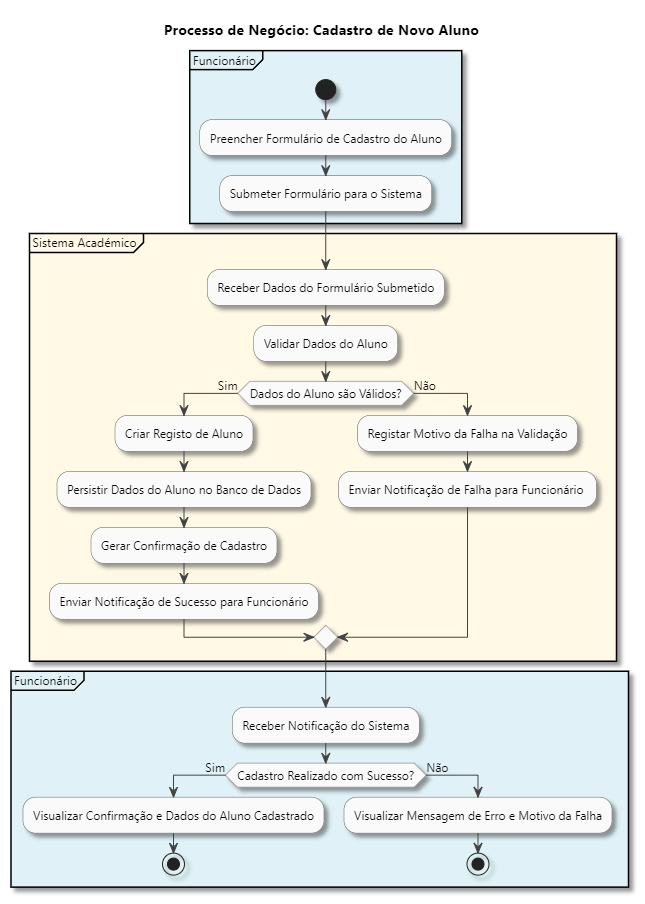
## **e) Manutenibilidade**

**[NF014] Código Bem Documentado**  
O código deve seguir padrões de codificação e estar devidamente comentado para facilitar a manutenção.  
**Prioridade:** ✔ Importante

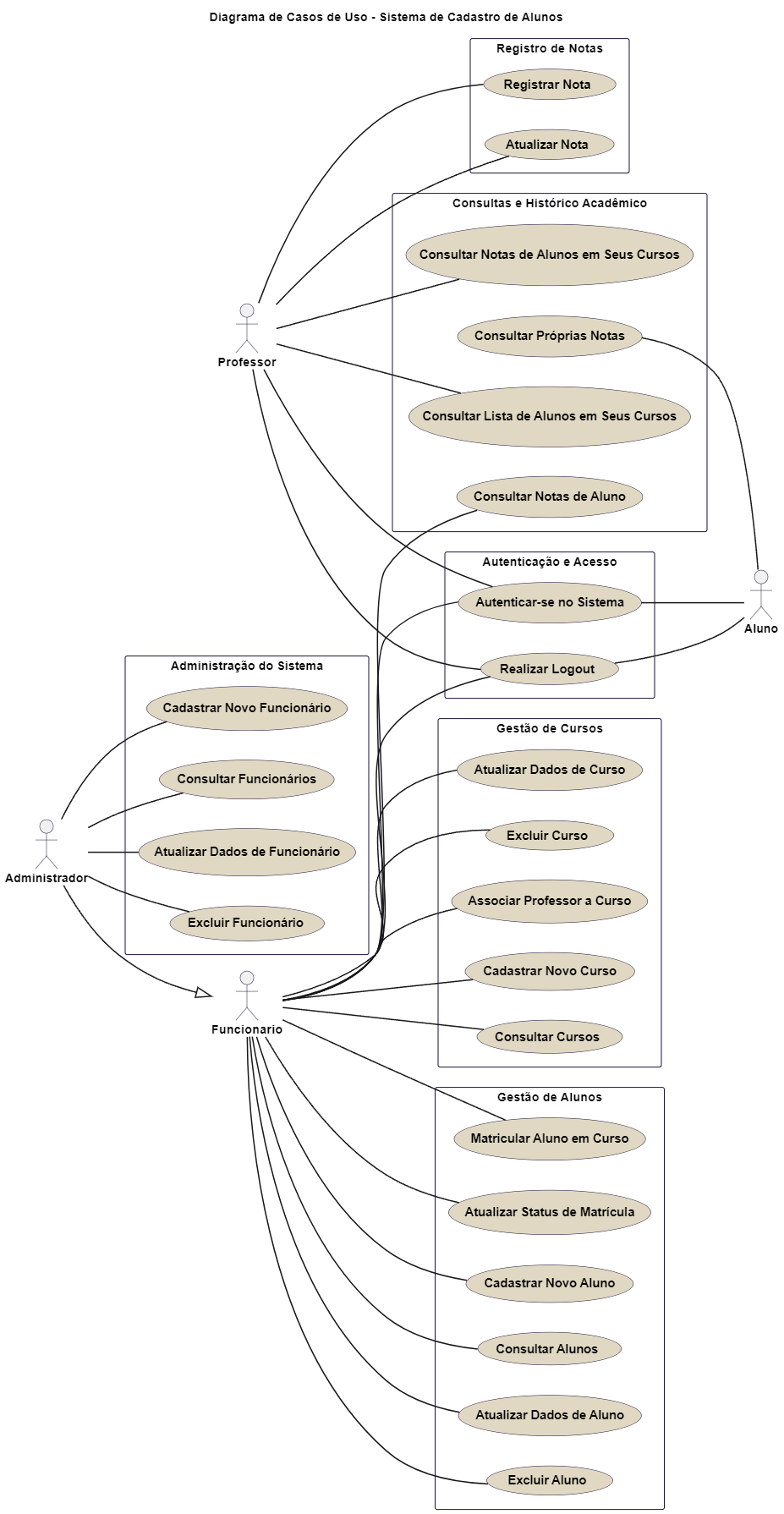
**[NF015] Arquitetura Modular**  
O sistema deve ser estruturado em camadas (Tela, Controlador, Serviço, Repositório), permitindo manutenção independente por módulo.  
**Prioridade:** ✔ Desejável

# Artefatos UML

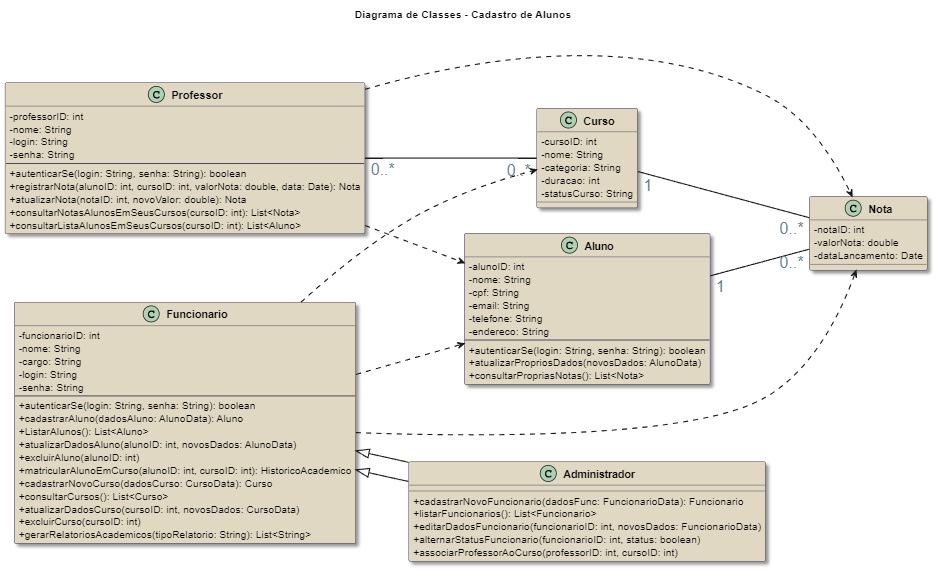
## Gestão de Processos de Negócio (BPM)



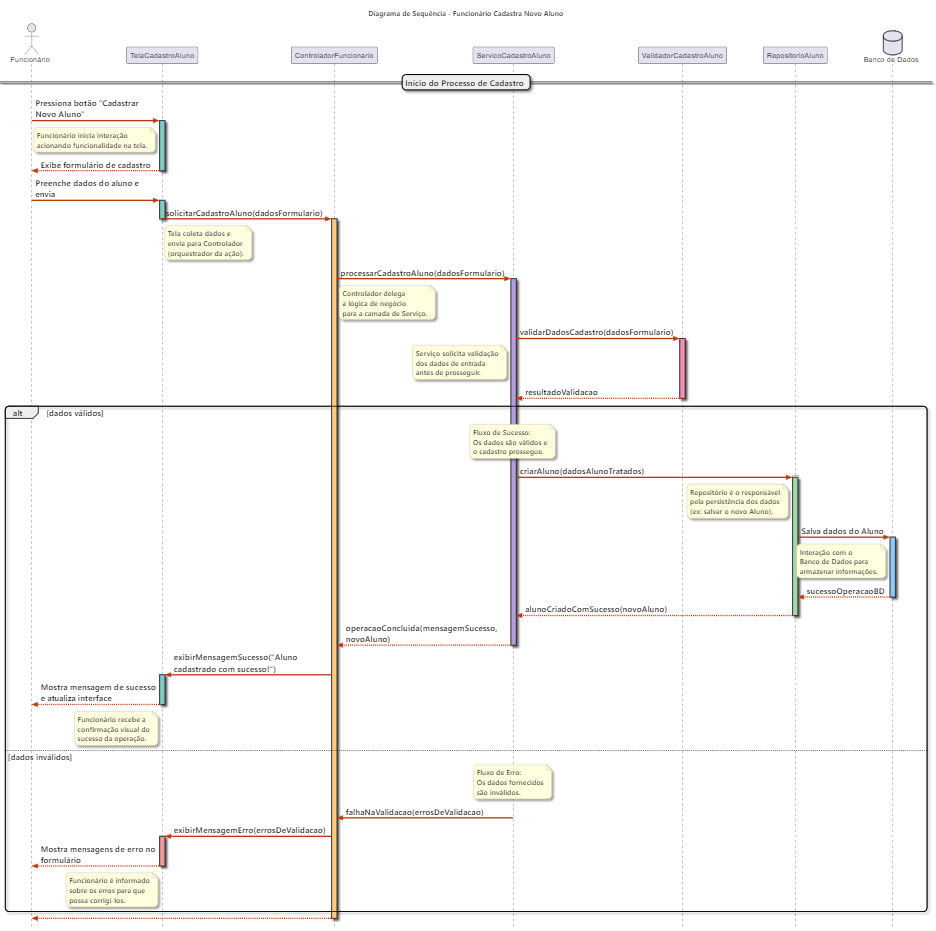
## Diagrama de Caso de Uso



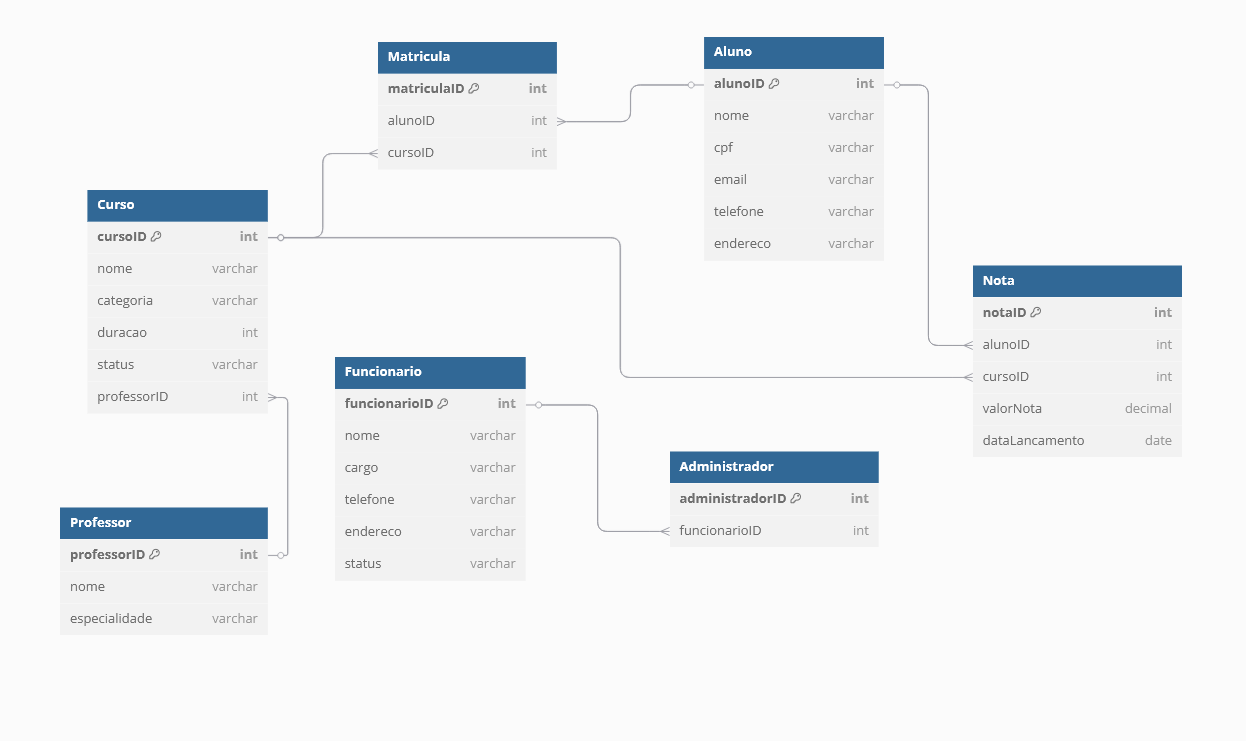
## Diagrama de Classe



## Diagrama de Sequência



## Diagrama Entidade Relacionamento - DER



# Banco de Dados

No sistema que desenvolvemos, usamos o MySQL como banco de dados para guardar todas as informações, como dados de alunos, funcionários, notas, cursos e administrador. Para conectar o sistema Java com o banco, utilizamos o JDBC, que é uma tecnologia que permite que o Java execute comandos SQL direto no banco, como inserir, atualizar, buscar e excluir registros. O MySQL é um banco de dados relacional, ou seja, ele organiza as informações em tabelas que se relacionam entre si. Criamos uma estrutura com tabelas para cada entidade do sistema, e elas são acessadas por meio de classes DAO no Java. Isso garantiu que os dados fossem bem organizados, fáceis de acessar e seguros dentro do sistema.

## 10.1 Relacionamento de tabelas

O sistema define seus relacionamentos de banco de dados no código CriadorTabela.java, que contém os comandos SQL responsáveis pela criação das tabelas e vínculos entre elas. Abaixo, são descritas as tabelas principais e seus relacionamentos:

### **Tabelas do Sistema**

* **funcionario**  
  Tabela base que armazena os dados dos funcionários.  
  Campos: funcionarioID, nome, cargo, login, senha, autenticado
* **professor**  
  Contém os dados dos professores.  
  Campos: professorID, nome, login, senha, autenticado
* **curso**  
  Armazena os cursos disponíveis e possui uma chave estrangeira para professorID, indicando o professor responsável.  
  Campos: cursoID, nome, categoria, duracao, professorID (FK)
* **aluno**  
  Registra os dados dos alunos.  
  Campos: alunoID, nome, cpf, email, telefone, endereco  
  (Não possui chaves estrangeiras diretas, mas se relaciona com outras tabelas por meio de notas e matrículas)
* **nota**  
  Registra as notas dos alunos nos cursos.  
  Campos: notaID, alunoID (FK), cursoID (FK), valorNota, dataLancamento
* **professor\_curso**  
  Tabela associativa com chave composta professorID e cursoID, permitindo o relacionamento muitos-para-muitos entre professores e cursos.
* **administrador**  
  Representa administradores que também são funcionários.  
  A chave primária administradorID também é chave estrangeira para funcionario.funcionarioID.

### **Relacionamentos**

* **Um professor** pode ministrar **vários cursos** → relação um-para-muitos (curso.professorID).
* **Um curso** pode ter **vários professores** → relação muitos-para-muitos via professor\_curso.
* **Um aluno** pode ter **múltiplas notas em diferentes cursos** → armazenadas na tabela nota.
* A tabela **administrador** atua como uma **especialização de funcionário**, pois compartilha a mesma chave primária.

Essa estrutura relacional garante integridade dos dados e facilita o cruzamento de informações no banco, permitindo consultas eficientes e consistência lógica entre as entidades do sistema.

# Conclusões

O Sistema de Cadastro de Alunos foi desenvolvido com o objetivo de aplicar conceitos fundamentais de engenharia de software, desde a definição de requisitos até a modelagem e implementação da solução. Ao longo do projeto, foram utilizados recursos como modelagem UML, arquitetura em camadas, banco de dados relacional (MySQL) e integração via JDBC.

A definição clara dos requisitos funcionais e não funcionais contribuiu para a organização do sistema e para a implementação orientada a perfis de usuários, garantindo controle de acesso e segurança. O uso de classes DAO facilitou a separação entre lógica de negócio e persistência de dados, promovendo maior organização e manutenção do código.

O sistema atendeu aos requisitos propostos e apresentou estrutura compatível com o ambiente acadêmico para o qual foi projetado. Como recomendações para versões futuras, sugere-se a implementação de uma interface gráfica, integração com serviços externos (como geração de relatórios) e ampliação das funcionalidades administrativas.

# Referências

Oracle. JDBC Overview. Disponível em: <https://docs.oracle.com/javase/8/docs/technotes/guides/jdbc/>  
MySQL. Documentação Oficial do MySQL. Disponível em: <https://dev.mysql.com/doc/>  
PlantUML. Guia de Diagramas. Disponível em: <https://plantuml.com/>  
GitHub Docs. Clonando repositórios. Disponível em: <https://docs.github.com/pt/repositories>  
Figma. Design de Interface de Usuário. Disponível em: <https://www.figma.com/>  
Análise de Requisitos. Modelos de Documentação. Disponível em: <https://analisederequisitos.com.br/>