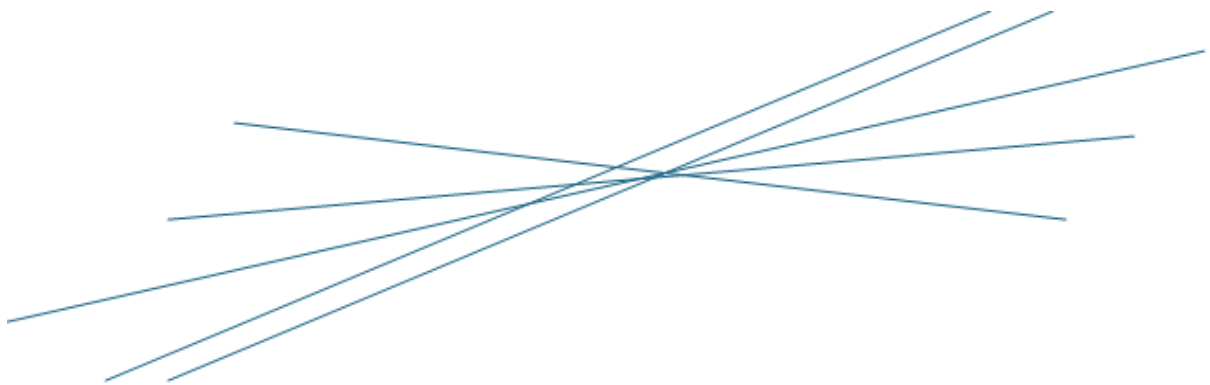


# PROJETO DE DESENVOLVIMENTO DE SOFTWARE

*Modelagem de Software*



Relatório técnico contendo Especificação  
do Software

2024

# Sumário

1. Introdução.....	3
2. Objetivo.....	5
3. Requisitos do Sistema.....	6
3.1 Requisitos Funcionais.....	6
3.2 Requisitos Não Funcionais.....	6
4. Diagramas em UML.....	7
4.1 Diagrama de Casos de Uso.....	7
4.2 Descrição dos Casos de Uso .....	8
4.3 Diagrama de Classes.....	9
4.4 Diagrama de Sequência.....	10
5. Modelo Entidade Relacionamento.....	11
6. Projeto de Banco de dados.....	12
6.1 Definição das Tabelas em SQL e funções.....	12
6.2 Descrição das Tabelas.....	18
6.3 Relacionamentos.....	18
7. Conclusão.....	19
8. Apresentação do Software.....	20

# Relatório Técnico de Especificação de Software

## 1. Introdução

Este relatório técnico apresenta a especificação detalhada de um software desenvolvido para atender às necessidades da Fundação Arte & Cultura Sempre. A fundação oferece disciplinas para crianças e adolescentes de até 16 anos, e atualmente enfrenta desafios significativos na gestão de matrículas devido ao uso de uma planilha eletrônica no Microsoft Excel. O aumento no número de estudantes e a frequência de matrículas em múltiplas disciplinas tornaram este método de controle inadequado e ineficiente.

Com o objetivo de solucionar esses problemas, a administração da fundação requisitou o desenvolvimento de um sistema informatizado que facilite o registro e a consulta dos dados de matrículas. Este novo sistema deve incluir funcionalidades que permitam:

- Registro da data de efetivação de cada matrícula, para fins de controle administrativo.
- Disponibilização de informações sobre o período de realização das disciplinas, indicando as datas de início e término, tanto para estudantes quanto para professores.

Para o desenvolvimento deste projeto, foi empregada uma abordagem de modelagem de software estruturada. A especificação aqui apresentada contempla uma série de componentes essenciais para a construção do sistema, que são:

- **Lista de requisitos funcionais e não funcionais:** Identificação detalhada das funcionalidades esperadas do sistema, bem como das restrições e padrões de qualidade que devem ser atendidos.
- **Diagrama de Casos de Uso:** Visualização das interações entre os usuários (atores) e o sistema, destacando os principais casos de uso.
- **Descrição do objetivo dos casos de uso:** Explicação detalhada do propósito e escopo de cada caso de uso identificado.
- **Diagrama de Classes:** Representação das classes e suas inter-relações dentro do sistema, fornecendo uma visão clara da estrutura de dados.
- **Diagrama de Sequência:** Ilustração do fluxo de interações necessárias para a funcionalidade de realização de matrícula em uma disciplina.

- **Modelo Entidade-Relacionamento:** Esquema representando as entidades de dados e seus relacionamentos, fundamental para a construção do banco de dados.
- **Projeto de Banco de Dados:** Desenho detalhado do banco de dados que suportará o sistema, incluindo tabelas, colunas e tipos de dados.

A especificação do sistema visa não apenas atender às necessidades imediatas da Fundação Arte & Cultura Sempre, mas também proporcionar uma base sólida para futuras expansões e melhorias, garantindo uma gestão mais eficiente e organizada das disciplinas oferecidas.

## 2. Objetivo

O objetivo deste projeto é desenvolver um sistema de software que atenda às necessidades administrativas da Fundação Arte & Cultura Sempre, melhorando significativamente o controle e a gestão dos dados de matrículas de suas disciplinas. Atualmente, o processo de registro de matrículas é realizado manualmente em uma planilha eletrônica, o que resulta em dificuldades de gerenciamento e riscos de erros devido ao crescente número de estudantes e à frequente inscrição em múltiplas disciplinas.

O novo sistema proposto tem como principais objetivos:

1. **Automatizar o Processo de Matrícula:** Eliminar a dependência de planilhas eletrônicas, proporcionando uma interface intuitiva e eficiente para registrar as matrículas dos estudantes.
2. **Registrar e Consultar Datas de Matrícula:** Incluir funcionalidades que permitam registrar a data de efetivação de cada matrícula, facilitando o controle administrativo e o acompanhamento de datas importantes.
3. **Disponibilizar Informações de Disciplinas:** Criar um meio acessível para que estudantes e professores possam consultar os períodos de realização das disciplinas, (datas de início e término).
4. **Melhorar a Gestão de Dados:** Garantir que todos os dados sejam armazenados de forma organizada e segura em um banco de dados, permitindo facilmente consulta e manipulação das informações.
5. **Facilitar a Consulta de Informações:** Proporcionar uma fonte confiável onde estudantes e professores possam consultar rapidamente os detalhes das disciplinas, incluindo horários e disponibilidade.
6. **Reduzir Erros e Redundâncias:** Minimizar os erros comuns em processos manuais e redundâncias de dados, assegurando que as informações sejam precisas e atualizadas.
7. **Preparar para Expansões Futuras:** Estruturar o sistema de forma flexível, possibilitando a adição de novas funcionalidades e o suporte a um número crescente de usuários e disciplinas.

Ao alcançar esses objetivos, o sistema contribuirá para uma administração mais eficiente e eficaz, permitindo que a fundação se concentre mais em sua missão educativa e cultural, com menos tempo gasto em tarefas administrativas.

## 3. Requisitos do Sistema

### 3.1 Requisitos Funcionais

**Cadastro de Aluno:** O sistema deve permitir o cadastro de informações de alunos, incluindo ID, nome, data de nascimento, nome do responsável, contatos.

**Cadastro de Disciplinas:** O sistema deve permitir o cadastro de informações de disciplinas, incluindo ID, nome, horário, data de início, data de término e professor responsável.

**Cadastro de Professor:** O sistema deve permitir o cadastro de informações de professores, incluindo ID, nome e disciplina que leciona.

**Matrícula de Aluno:** O sistema deve permitir a matrícula de alunos em disciplinas, registrando a data de efetivação da matrícula.

**Consulta de Datas das Disciplinas:** O sistema deve permitir que estudantes e professores consultem as datas de início e término das disciplinas ao longo do ano.

**Consulta de Dados pelo Administrador:** O sistema deve permitir que o administrador consulte todos os dados relacionados ao sistema e realiza cadastros tanto da matrícula quanto de alunos e professores.

### 3.2 Requisitos Não Funcionais

**Usabilidade:** O sistema deve ser intuitivo e de fácil utilização, mesmo por usuários sem experiência técnica.

**Desempenho:** O sistema deve ter um tempo de resposta rápido para consultas e atualizações de dados.

**Segurança:** O acesso ao sistema deve ser controlado por meio de autenticação, garantindo que apenas usuários autorizados possam realizar operações (Administrador).

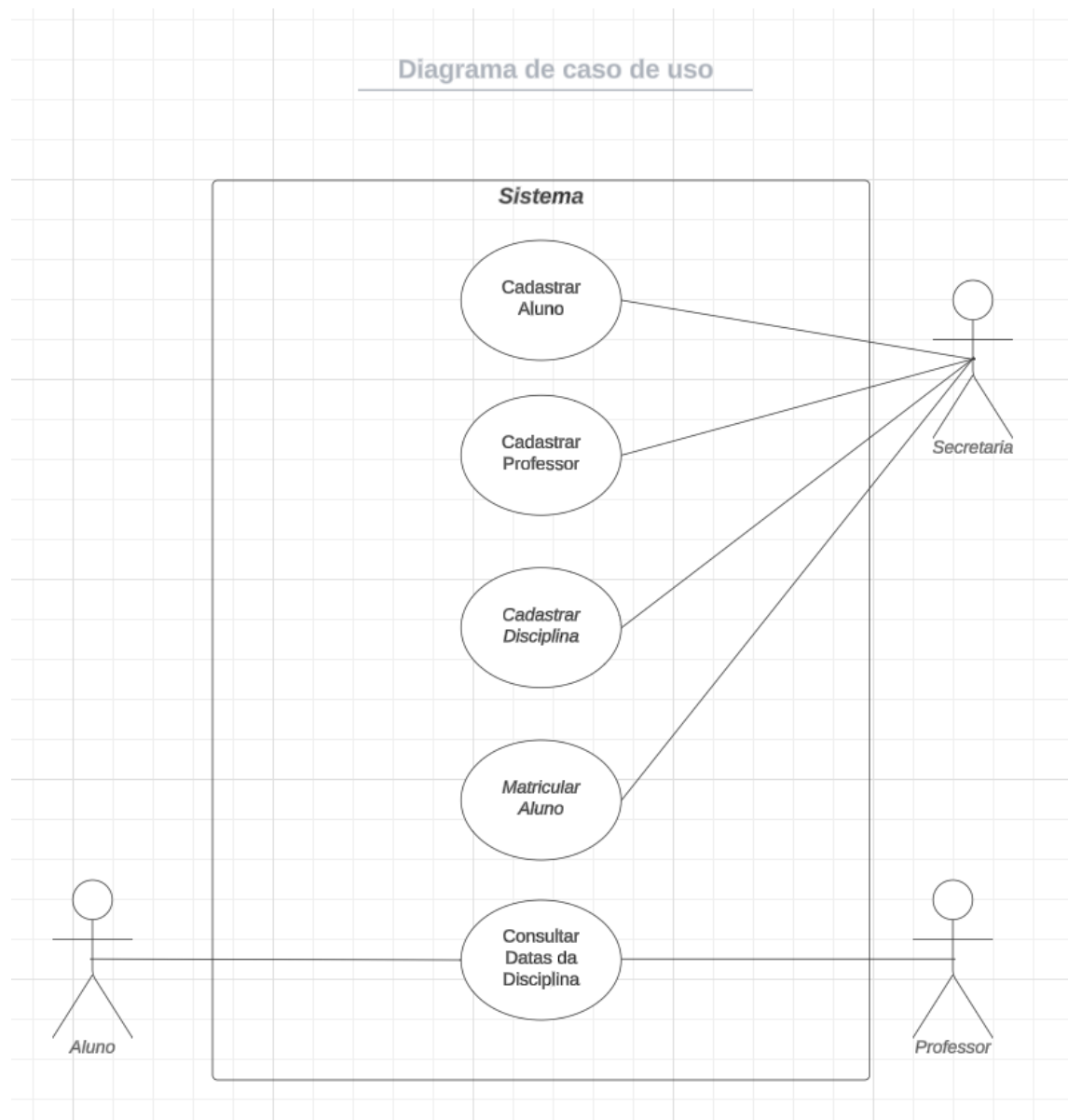
**Disponibilidade:** O sistema deve estar disponível para acesso a maior parte do tempo, com um mínimo de interrupções para manutenção.

**Escalabilidade:** O sistema deve ser capaz de lidar com um aumento no número de usuários e de registros sem comprometer o desempenho.

Compatibilidade: O sistema deve ser compatível com diferentes dispositivos e navegadores, garantindo uma boa experiência de uso em qualquer plataforma.

## 4. Diagramas em UML

### 4.1 Diagrama de Casos de Uso



## 4.2 Descrição dos Casos de Uso

Atores: Aluno, Secretária, Professor.

Funcionalidades:

Cadastrar Aluno: Permite a secretária a cadastrar um novo aluno ao sistema.

Cadastrar Professor: Permite a secretária cadastrar professores no sistema, que irão lecionar nas disciplinas cadastradas no sistema.

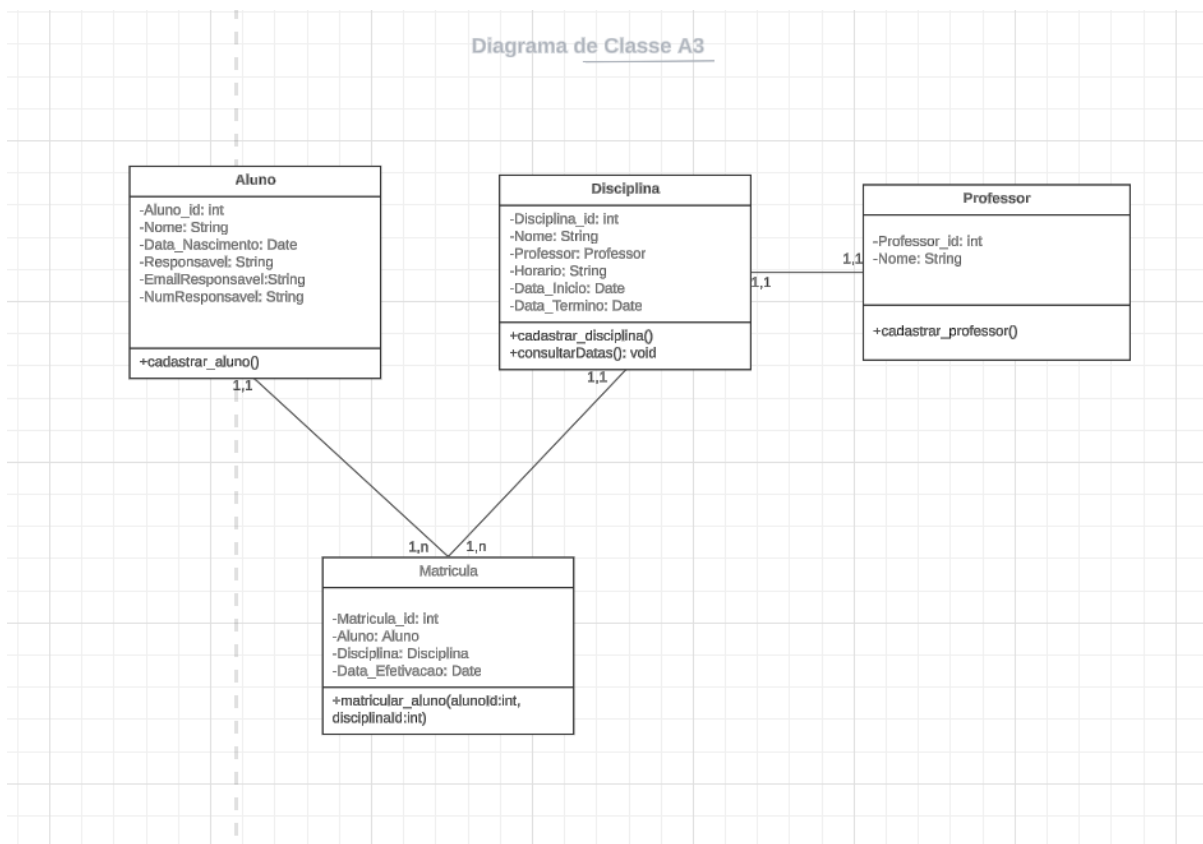
Cadastrar Disciplina: Permite que a secretária (Administrador) cadastre disciplinas no sistema, com informações de: nome, horário, data de início, data de término e professor responsável.

Matricular Aluno: Permite a secretária(administrador) a matricular um aluno em uma disciplina e assim registrar a data de efetivação da matrícula.

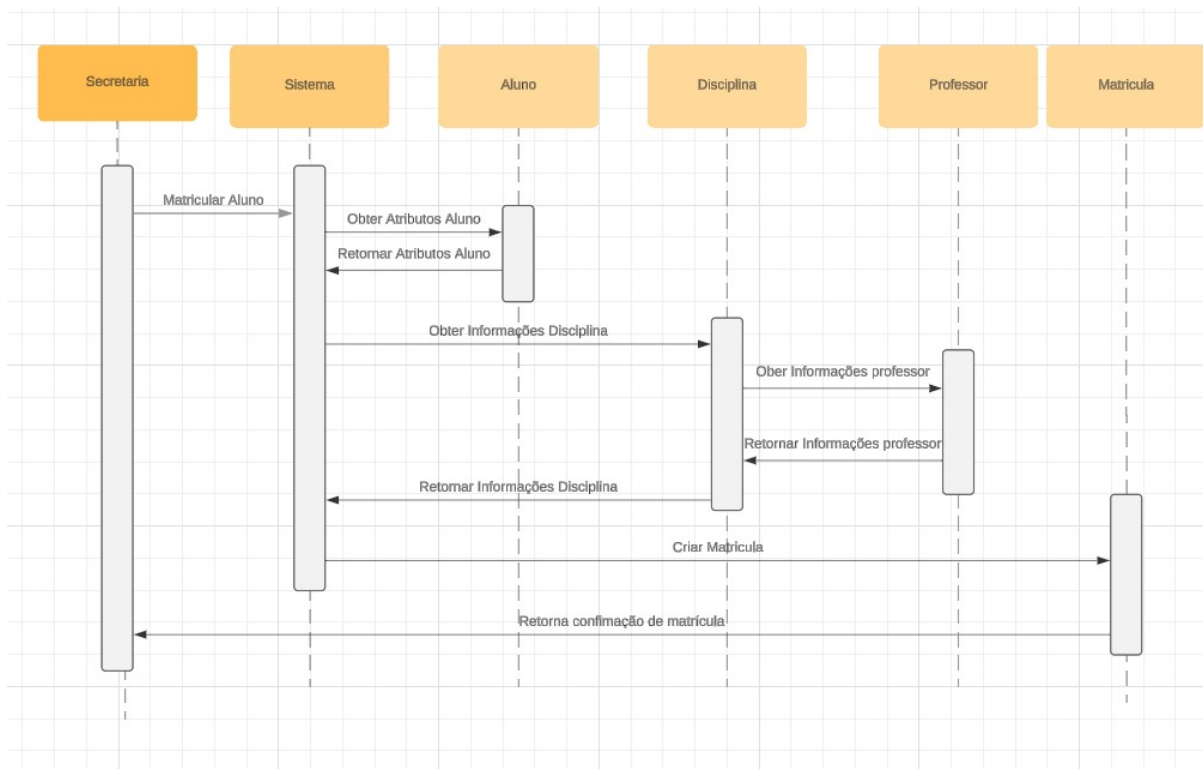
Consultar Período da Disciplina: Permitir que alunos e professores consigam consultar datas de início e término das disciplinas ao longo do ano.



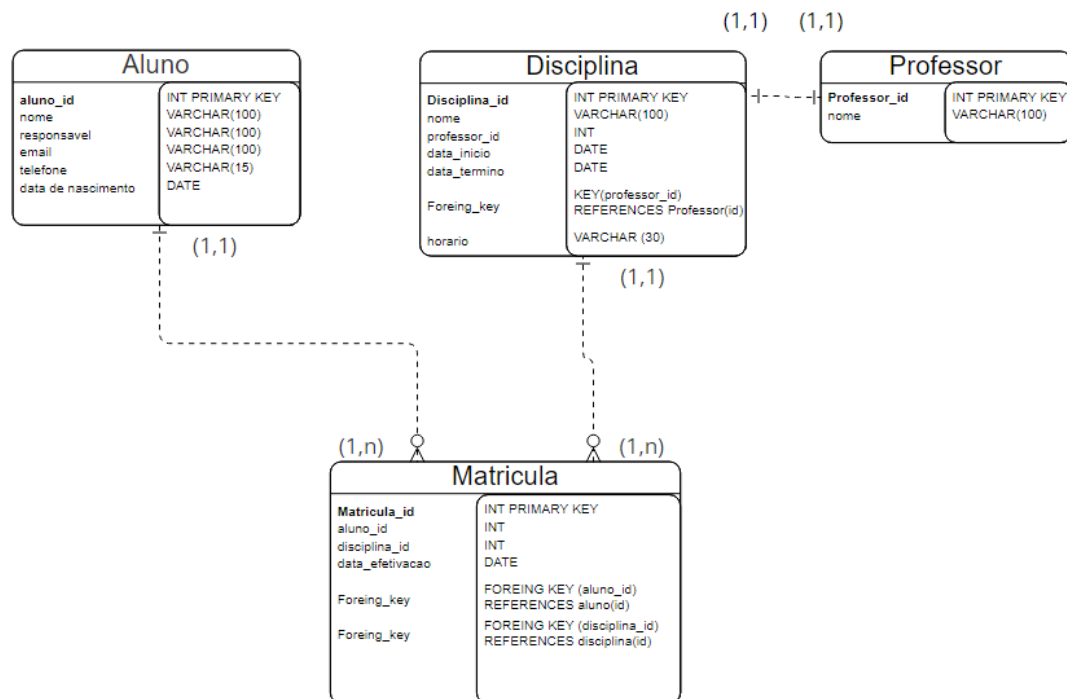
## 4.3 Diagrama de Classes



## 4.4 Diagrama de Sequência



## 5. Modelo Entidade Relacionamento



## 6. Projeto de Banco de dados

Para o sistema da Fundação Arte & Cultura Sempre, o banco de dados deve ser bem estruturado para suportar todas as funcionalidades necessárias, incluindo o registro de matrículas, consulta de informações de cursos, e gerenciamento de alunos e professores. A seguir, apresento um exemplo de projeto de banco de dados com as principais tabelas e suas relações.

## 6.1 Definição das Tabelas em SQL e funções

Abaixo, forneço o script SQL para criar as tabelas do banco de dados e funções para matricular aluno e consultar data de início e término da disciplina.

-- Criação das tabelas

CREATE TABLE Aluno (

Aluno\_id INT PRIMARY KEY AUTO\_INCREMENT,

Nome VARCHAR (100) NOT NULL,

Data\_Nascimento DATE NOT NULL,

Responsavel VARCHAR (100),

EmailResponsavel VARCHAR (100),

NumResponsavel VARCHAR (15)

);

CREATE TABLE Professor (

Professor\_id INT PRIMARY KEY AUTO\_INCREMENT,

Nome VARCHAR (100) NOT NULL

);

```

CREATE TABLE Disciplina (

    Disciplina_id INT PRIMARY KEY AUTO_INCREMENT,

    Nome VARCHAR (100) NOT NULL,

    Horario VARCHAR (30),

    Data_Inicio DATE,

    Data_Termino DATE,

    Professor_id INT,

    FOREIGN KEY (Professor_id) REFERENCES Professor (Professor_id)

);

```

```

CREATE TABLE Matricula (

    Matricula_id INT PRIMARY KEY AUTO_INCREMENT,

    Aluno_id INT,

    Disciplina_id INT,

    Data_Efetivacao DATE,

    FOREIGN KEY (Aluno_id) REFERENCES Aluno (Aluno_id),

    FOREIGN KEY (Disciplina_id) REFERENCES Disciplina (Disciplina_id)

);

```

```

-- Função para cadastrar aluno

```

```

DELIMITER //

```

```

CREATE PROCEDURE cadastrar_aluno(

```

```

    IN p_Nome VARCHAR (100),

```

```

    IN p_Data_Nascimento DATE,

    IN p_Responsavel VARCHAR (100),

    IN p_EmailResponsavel VARCHAR (100),

    IN p_NumResponsavel VARCHAR (15)

)

BEGIN

    INSERT INTO Aluno (Nome, Data_Nascimento, Responsavel, EmailResponsavel, NumResponsavel)

    VALUES (p_Nome, p_Data_Nascimento, p_Responsavel, p_EmailResponsavel, p_NumResponsavel);

END //

DELIMITER;

-- Exemplo de chamada

CALL cadastrar_aluno('João da Silva', '2000-01-15', 'Josefa da Silva', 'josefa@example.com',
'11923456789');

-- Função para cadastrar professor

DELIMITER //

CREATE PROCEDURE cadastrar_professor(

    IN p_Nome VARCHAR(100)

)

BEGIN

    INSERT INTO Professor (Nome)

    VALUES (p_Nome);

END //

DELIMITER ;

```

-- Exemplo de chamada

```
CALL cadastrar_professor('Victor Crispim');
```

-- Função para cadastrar disciplina

DELIMITER //

```
CREATE PROCEDURE cadastrar_disciplina(
```

```
    IN p_Nome VARCHAR(100),
```

```
    IN p_Horario VARCHAR(30),
```

```
    IN p_Data_Inicio DATE,
```

```
    IN p_Data_Termino DATE,
```

```
    IN p_Professor_id INT
```

```
)
```

```
BEGIN
```

```
-- Verifica se o professor existe
```

```
IF (SELECT COUNT(*) FROM Professor WHERE Professor_id = p_Professor_id) = 0 THEN
```

```
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Professor não encontrado';
```

```
END IF;
```

```
INSERT INTO Disciplina (Nome, Horario, Data_Inicio, Data_Termino, Professor_id)
```

```
VALUES (p_Nome, p_Horario, p_Data_Inicio, p_Data_Termino, p_Professor_id);
```

```
END //
```

```
DELIMITER ;
```

-- Exemplo de chamada

```

-- Supondo que o professor com id 1 seja o responsável pela disciplina

CALL cadastrar_disciplina('Matemática', '08:00 - 10:00', '2024-09-01', '2024-12-15', 1);


-- Função matricular aluno

DELIMITER //

CREATE PROCEDURE matricular_aluno(

    IN p_Aluno_id INT,

    IN p_Disciplina_id INT,

    IN p_data_efetivacao DATE

)

BEGIN

    -- Verifica se o aluno existe

    IF (SELECT COUNT(*) FROM Aluno WHERE Aluno_id = p_Aluno_id) = 0 THEN

        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Aluno não encontrado';

    END IF;

    -- Verifica se a disciplina existe

    IF (SELECT COUNT(*) FROM Disciplina WHERE Disciplina_id = p_Disciplina_id) = 0 THEN

        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Disciplina não encontrada';

    END IF;

    -- Insere a matrícula

    INSERT INTO Matricula (Aluno_id, Disciplina_id, Data_Efetivacao)

    VALUES (p_Aluno_id, p_Disciplina_id, p_data_efetivacao);

END //

```



```
DELIMITER ;
```

```
-- Exemplo de chamada
```

```
CALL matricular_aluno(1, 2, '2024-06-01');
```

```
-- Função para consultar datas da disciplina
```

```
DELIMITER //
```

```
CREATE PROCEDURE consultarDatas(
```

```
    IN p_Disciplina_id INT
```

```
)
```

```
BEGIN
```

```
    SELECT Nome, Data_Inicio, Data_Termino
```

```
    FROM Disciplina
```

```
WHERE Disciplina_id = p_Disciplina_id;
```

```
END //
```

```
DELIMITER ;
```

```
-- Exemplo de chamada
```

```
CALL consultarDatas(2);
```

## 6.2 Descrição das Tabelas

- **Aluno:** Armazena informações sobre os alunos, incluindo dados pessoais e de contato.
- **Professor:** Contém dados sobre os professores.
- **Disciplina:** Guarda detalhes sobre as disciplinas oferecidas, incluindo datas e o professor responsável.
- **Matrícula:** Registra cada matrícula efetuada, associando aluno a disciplina e armazenando a data de efetivação.

## 6.3 Relacionamentos

- Cada aluno pode possuir várias matrículas, onde, cada matrícula possui sua própria disciplina.
- Cada disciplina é ministrada por um professor (Relacionamento de um-para-um).
- Cada matrícula associa um aluno a uma disciplina específica, registrando a data de efetivação.

Este projeto de banco de dados oferece uma base sólida para o sistema proposto, permitindo uma gestão eficiente das matrículas e disciplinas na Fundação Arte & Cultura Sempre.

## 7. Conclusão

Este relatório técnico apresentou a especificação detalhada para o desenvolvimento de um sistema de software destinado a otimizar a gestão das matrículas na Fundação Arte & Cultura Sempre. A análise inicial identificou diversos desafios enfrentados pela administração, que atualmente utiliza uma planilha eletrônica para registrar as matrículas dos alunos. Esses desafios incluem a dificuldade de controle e consulta dos dados, especialmente devido ao grande número de alunos e à frequência com que se inscrevem em múltiplas disciplinas.

Para resolver esses problemas, foi proposto um sistema informatizado abrangente que não só automatiza o processo de matrícula, mas também melhora a gestão e consulta dos dados. A especificação detalhada incluiu a definição de requisitos funcionais e não funcionais, diagramas de casos de uso e classes, diagramas de sequência para a funcionalidade de matrícula, e um modelo entidade-relacionamento para o projeto do banco de dados.

Os principais benefícios esperados com a implementação deste sistema incluem:

- 1) **Melhoria na Eficiência Administrativa:** A automação do processo de matrícula reduzirá o tempo e esforço necessários para gerenciar os registros dos alunos.
- 2) **Precisão e Confiabilidade dos Dados:** A eliminação do uso de planilhas manuais minimizará os erros e redundâncias, garantindo que as informações sejam precisas e atualizadas.
- 3) **Facilidade de Consulta:** Alunos e professores terão acesso fácil e rápido às informações sobre as disciplinas, incluindo datas de início e término, facilitando o planejamento e a participação.
- 4) **Flexibilidade para Expansões Futuras:** A estrutura modular e bem planejada do sistema permitirá que novas funcionalidades sejam adicionadas conforme necessário, suportando o crescimento contínuo da fundação.

Em resumo, o sistema proposto representa uma solução robusta e eficiente para os desafios atuais enfrentados pela Fundação Arte & Cultura Sempre. Sua implementação não apenas simplificará a gestão administrativa, mas também proporcionará uma melhor experiência para alunos e professores, alinhando-se com a missão da fundação de promover educação e cultura de forma acessível e organizada. Este projeto, desenvolvido com base em técnicas modernas de modelagem de software, prepara o caminho para um futuro mais eficiente e gerenciável na administração das disciplinas oferecidas pela fundação.

## 8. Apresentação do software

Para concluir o projeto segue link do vídeo apresentando o software e suas vantagens.

[https://drive.google.com/file/d/1Myd1gMjRExNQEUG\\_iXoD6t56CfehkJat/view?usp=sharing](https://drive.google.com/file/d/1Myd1gMjRExNQEUG_iXoD6t56CfehkJat/view?usp=sharing)