

CENTRO UNIVERSITÁRIO SENAC

**PROJETO INTEGRADOR: DESENVOLVIMENTO DE SISTEMAS ORIENTADO A
OBJETOS**

2ª ENTREGA DO PROJETO INTEGRADOR

São Paulo

2025

CENTRO UNIVERSITÁRIO SENAC

Disciplina: Projeto Integrador

**PROJETO INTEGRADOR: DESENVOLVIMENTO DE SISTEMAS ORIENTADO A
OBJETOS**

Delfino Maurício Cândido
José Márcio Canto Oliveira
Pedro Henrique Beltrão de Souza
Renan Telo de Moraes
Vinicius Luscri

**São Paulo
2025**

SUMÁRIO

1. INTRODUÇÃO	4
2. OBJETIVOS DO PROJETO	4
2.1. Objetivo Geral	4
2.2. Objetivos Específicos	4
3. ESTRUTURA DO PROJETO.....	4
3.1. Organização de Arquivos	5
4. PROTÓTIPOS DESENVOLVIDOS	5
4.1. Características Técnicas	5
5. BANCO DE DADOS	6
5.1. Scripts DDL (Data Definition Language)	6
5.2. Scripts DML (Data Manipulation Language).....	6
6. DIAGRAMAS UML.....	7
6.1. Diagrama de Caso de Uso	7
6.2. Diagrama de Classes	7
7. TECNOLOGIAS UTILIZADAS	7
8. CONSIDERAÇÕES FINAIS.....	8
8.1. Resultados Obtidos	8
8.2. Aprendizados	8
8.3. Link Para Github:	8

1. INTRODUÇÃO

Este relatório apresenta a segunda entrega do Projeto Integrador da disciplina de Desenvolvimento de Sistemas Orientado a Objetos. O projeto consiste no desenvolvimento de um Sistema de Gestão Universitária, que tem como objetivo gerenciar cadastros de diferentes tipos de pessoas que interagem com uma instituição de ensino superior.

O sistema foi desenvolvido aplicando os conceitos fundamentais de Orientação a Objetos, incluindo herança, polimorfismo, encapsulamento e abstração. A modelagem foi realizada utilizando diagramas UML (Unified Modeling Language), e a implementação inclui protótipos funcionais das interfaces e scripts SQL para persistência de dados.

2. OBJETIVOS DO PROJETO

2.1. Objetivo Geral

Desenvolver um sistema de gestão universitária que permita o cadastro e gerenciamento de diferentes tipos de pessoas relacionadas à instituição de ensino, aplicando conceitos de programação orientada a objetos e modelagem de dados.

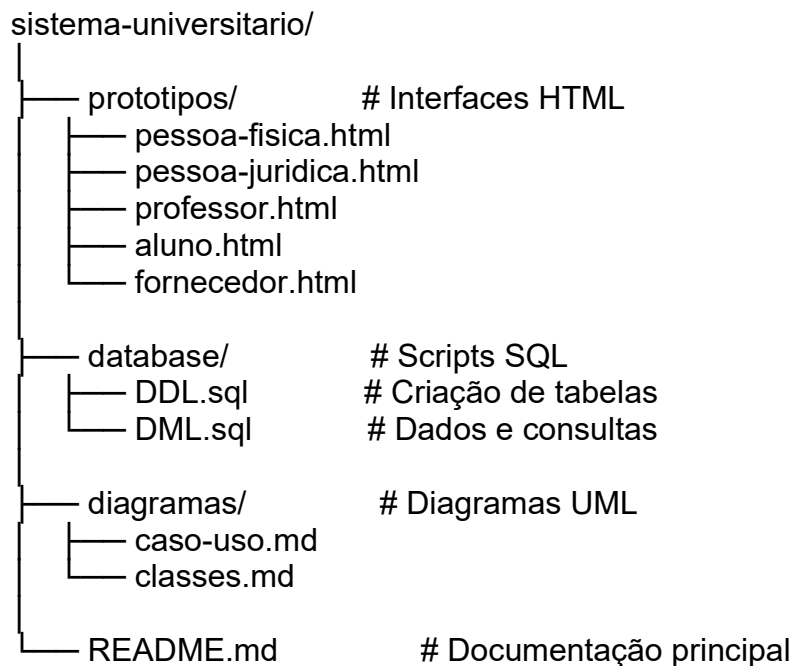
2.2. Objetivos Específicos

- Criar protótipos funcionais para 5 jornadas de cadastro
- Desenvolver modelo de dados relacional com scripts SQL
- Implementar herança de classes no banco de dados
- Documentar o projeto utilizando Markdown
- Organizar o projeto em repositório GitHub

3. ESTRUTURA DO PROJETO

O projeto foi organizado seguindo boas práticas de desenvolvimento, com separação clara entre protótipos, scripts de banco de dados e documentação.

3.1. Organização de Arquivos



4. PROTÓTIPOS DESENVOLVIDOS

Foram desenvolvidos 5 protótipos funcionais em HTML/CSS, cada um representando uma jornada de cadastro diferente. Os protótipos são totalmente funcionais e podem ser abertos diretamente no navegador.

PROTÓTIPO	ARQUIVO	CAMPOS PRINCIPAIS
Pessoa Física	pessoa-fisica.html	CPF, Nome, Data Nascimento, Email, Telefone, Endereço
Pessoa Jurídica	pessoa-juridica.html	CNPJ, Razão Social, Nome Fantasia, Email, Telefone
Professor	professor.html	CPF, Nome, Matrícula, Departamento, Titulação
Aluno	aluno.html	CPF, Nome, Matrícula, Curso, Turno, Período
Fornecedor	fornecedor.html	CNPJ, Razão Social, Setor, Produtos/Serviços

4.1. Características Técnicas

- HTML5: Estrutura semântica e validação de formulários
- CSS3: Design responsivo e estilização moderna

- Validação: Campos obrigatórios e tipos de dados adequados
- UX: Interface intuitiva e amigável

5. BANCO DE DADOS

O banco de dados foi modelado para refletir a hierarquia de classes da programação orientada a objetos, implementando o conceito de herança através de chaves estrangeiras.

5.1. Scripts DDL (Data Definition Language)

O script DDL contém a criação do banco de dados e de 6 tabelas que implementam a hierarquia de herança:

TABELA	DESCRIÇÃO	CAMPOS PRINCIPAIS
pessoa	Classe base abstrata	email, telefone, endereço, cidade, estado, cep
pessoa_fisica	Herda de pessoa	cpf, nome, data_nascimento + FK pessoa
pessoa_juridica	Herda de pessoa	cnpj, razao_social, nome_fantasia + FK pessoa
professor	Herda de pessoa_fisica	matricula, departamento, titulacao + FK pessoa_fisica
aluno	Herda de pessoa_fisica	matricula, curso, turno, periodo + FK pessoa_fisica
fornecedor	Herda de pessoa_juridica	setor_atuacao, produtos_servicos + FK pessoa_juridica

5.2. Scripts DML (Data Manipulation Language)

O script DML contém comandos para manipulação dos dados:

- INSERT: Inserção de 7 registros de exemplo (3 pessoas físicas, 2 jurídicas, 1 professor, 2 alunos, 2 fornecedores)
- SELECT: 3 consultas com JOIN para recuperar dados completos
- UPDATE: 3 exemplos de atualização de registros
- DELETE: Exemplos comentados de exclusão

6. DIAGRAMAS UML

6.1. Diagrama de Caso de Uso

O diagrama de caso de uso identifica o ator principal (Administrador) e os 5 casos de uso do sistema. Cada caso de uso representa uma jornada de cadastro diferente.

Ator: Administrador

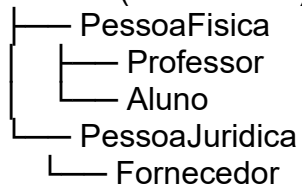
- 1 - Cadastrar Pessoa Física
- 2 - Cadastrar Pessoa Jurídica
- 3 - Cadastrar Professor
- 4 - Cadastrar Aluno
- 5 - Cadastrar Fornecedor

6.2. Diagrama de Classes

O diagrama de classes mostra a hierarquia de herança implementada no sistema:

Hierarquia de Herança:

Pessoa (classe base)



7. TECNOLOGIAS UTILIZADAS

TECNOLOGIA	USO NO PROJETO
HTML5	Estrutura dos protótipos de interface
CSS3	Estilização e design responsivo
MySQL	Sistema de gerenciamento de banco de dados
Markdown	Documentação do projeto e diagramas
Git/GitHub	Controle de versão e colaboração

8. CONSIDERAÇÕES FINAIS

Este projeto integrador permitiu a aplicação prática dos conceitos fundamentais de Orientação a Objetos e modelagem de sistemas. A segunda entrega contemplou o desenvolvimento de protótipos funcionais e a implementação do modelo de dados relacional, demonstrando a integração entre modelagem UML e desenvolvimento prático.

Os objetivos estabelecidos foram alcançados: foram desenvolvidos 5 protótipos funcionais conforme especificado, os scripts SQL (DDL e DML) foram implementados refletindo a hierarquia de classes modelada, e toda a documentação foi organizada em formato Markdown no repositório GitHub.

8.1. Resultados Obtidos

- 5 protótipos HTML funcionais e responsivos
- Banco de dados com 6 tabelas implementando herança
- Scripts SQL completos (DDL + DML)
- Documentação completa em Markdown
- Diagramas UML (Caso de Uso e Classes)
- Projeto organizado no GitHub

8.2. Aprendizados

O desenvolvimento deste projeto proporcionou experiência prática em:

- Modelagem de sistemas usando UML
- Implementação de herança em banco de dados relacional
- Desenvolvimento de interfaces web
- Trabalho em equipe e uso de ferramentas de colaboração
- Documentação técnica de projetos

8.3. Link Para Github:

<<https://github.com/candidodm/Projeto-Integrador---2-Entrega---Sistema-de-Gest-o-Universit-ria---Grupo-48.git>>