

12 DE OUTUBRO DE 2025

DOMAIN DRIVEN DESIGN USING JAVA

SPRINT [1] REPORT



Turma ITDSPR
RM 568542 - Hugo Souza de Jesus
RM 566815 - Lucas Campanhã dos Santos
RM 567010 - Lucas Marcelino Pompeu

Tech
do
Bem 

SUMÁRIO (EXEC) {

DESCRITIVO_DO_PROJETO

PAG_3 >> 1.1. CONTEXTO DO PROJETO

PAG_3 >> 1.2. OBJETIVOS

PAG_3 >> 1.2.1. OBJETIVO GERAL

PAG_4 >> 1.2.2. OBJETIVOS ESPECÍFICOS

PAG_4 >> 1.3. JUSTIFICATIVA

MODELAGEM_DAS_CLASSES_{UML};

PAG_5 >> 2.1. PADRÕES E CONVENÇÕES ADOTADAS

PAG_6 >> 2.2. DESCRIÇÃO DAS CLASSES

PAG_6 >> 2.2.1. CLASSE P/ ENDEREÇO

PAG_6 >> 2.2.2. CLASSE P/ PARCEIRO

PAG_7 >> 2.2.3. CLASSE P/ DOADOR

PAG_7 >> 2.2.4. CLASSE P/ PROGRAMA

PAG_8 >> 2.2.5. CLASSE P/ VOLUNTÁRIO

PAG_9 >> 2.2.6. CLASSE P/ PACIENTE

PAG_9 >> 2.2.7. CLASSE P/ CAMPANHA

PAG_10 >> 2.2.8. CLASSE P/ DOAÇÃO

PAG_11 >> 2.2.9. CLASSE P/ CONSULTA

PAG_11 >> 2.2.10. CLASSE P/ FINANCEIRO

PAG_12 >> 2.2.11. CLASSE P/ CANAL DE COMUNICAÇÃO

}

DESCRITIVO _DO_PROJETO

1.1. CONTEXTO DO PROJETO

Este documento detalha a estrutura de classes de um Sistema de Gestão para Organizações Sociais (SGO), projetado para centralizar e otimizar as operações de uma entidade sem fins lucrativos. A solução visa gerenciar de forma integrada os múltiplos relacionamentos e processos que compõem o ecossistema de uma ONG, como o cadastro de voluntários e pacientes, a organização de programas e campanhas de arrecadação, o registro de doações e a gestão de consultas ou atendimentos. O modelo foi concebido para ser robusto e escalável, servindo como a espinha dorsal para uma futura aplicação a ser desenvolvida em Java. A arquitetura proposta organiza as informações de forma coesa, permitindo que a organização tenha uma visão 360 graus de suas atividades, desde o planejamento financeiro até a execução de seus projetos sociais e o impacto gerado junto aos beneficiários. O sistema proposto reflete a complexidade do terceiro setor, tratando cada stakeholder (parceiro, doador, voluntário e paciente) como uma entidade distinta, mas interconectada dentro de um fluxo de trabalho unificado, garantindo a integridade e a rastreabilidade dos dados.

1.2. OBJETIVOS

1.2.1. OBJETIVO GERAL:

Modelar uma estrutura de software orientada a objetos que sirva de base para um sistema de gestão completo, capaz de centralizar o controle de todas as frentes de atuação de uma organização social, melhorando a eficiência operacional e a transparência.

1.2.2. OBJETIVOS ESPECÍFICOS:

- A. Estruturar o cadastro e gerenciamento de todos os envolvidos: Pacientes, Voluntários, Doadores e Parceiros.
- B. Modelar a organização de iniciativas, como Programas e Campanhas, e seus respectivos vínculos.
- C. Permitir o registro e controle de transações financeiras, como Doações e Despesas gerais.
- D. Estruturar o agendamento e o histórico de atendimentos, como Consultas.
- E. Garantir a integridade e o reuso de dados, como informações de Endereço.

1.3. JUSTIFICATIVA

Organizações do terceiro setor lidam com um grande volume de informações heterogêneas, desde dados cadastrais de pessoas até controles financeiros e logísticos de campanhas. A ausência de um sistema centralizado frequentemente leva à duplicação de dados, dificuldade na geração de relatórios de impacto e ineficiência na comunicação entre as equipes. Este projeto se justifica pela necessidade de prover uma solução tecnológica que enderece esses desafios. Ao modelar as classes de forma clara e inter-relacionada, criamos um alicerce sólido para uma aplicação que não apenas organizará os dados, mas também potencializará a capacidade da ONG de gerir seus recursos, otimizar o trabalho dos voluntários e, conseqüentemente, ampliar o alcance e a qualidade de seus serviços prestados à comunidade.

MODELAGEM_DAS _CLASSES_{UML}

2.1. PADRÕES E CONVENÇÕES ADOTADAS

A modelagem foi realizada seguindo rigorosamente os padrões da UML (Unified Modeling Language) para diagramas de classes.

Número de Classes: O projeto é composto por 11 classes, superando o mínimo de 10 exigido.

Encapsulamento: Todos os atributos de todas as classes são declarados com visibilidade privada (-). Esta decisão de design garante o encapsulamento dos dados, protegendo o estado dos objetos e permitindo que o acesso seja controlado por meio de métodos públicos (getters e setters), em conformidade com as boas práticas de programação orientada a objetos em Java.

Nomenclatura e Tipos: A nomenclatura segue o padrão da linguagem Java, com nomes de classes em PascalCase e nomes de atributos em camelCase. Os tipos de dados (String, int, double, boolean) são compatíveis com os tipos primitivos e de referência da linguagem, e as associações entre objetos são representadas por atributos cujo tipo é outra classe do diagrama.

Parceiro
<ul style="list-style-type: none">- id : String- nome : String- contato : String- endereco : Endereco

Doador
<ul style="list-style-type: none">- id : String- nome : String- contato : String- endereco : Endereco

Voluntario
<ul style="list-style-type: none">- id : String- nome : String- profissão : String- contato : String- programasAtuacao : Programa- endereco : Endereco

Paciente
<ul style="list-style-type: none">- id : String- nome : String- idade : int- contato : String- endereco : Endereco- historico : String- programaVinculado : Programa

Programa
<ul style="list-style-type: none">- id : String- nome : String- descrição : String- campanhaVinculada : Campanha

Campanha
<ul style="list-style-type: none">- id : String- nome : String- objetivo : String- metaFinanceira : double- programaVinculado : Programa

Doacao
<ul style="list-style-type: none">- id : String- valor : double- data : String- campanha : Campanha- doador : Doador

Endereco
<ul style="list-style-type: none">- logradouro : String- numero : int- complemento : String- bairro : String- cidade : String- estado : String

Consulta
<ul style="list-style-type: none">- id : String- data : String- requisitos : String- resultado : String- paciente : Paciente- dentista : Voluntario

Financeiro
<ul style="list-style-type: none">- id : String- anoFiscal : int- receitaTotal : double- despesas : double

CanalComunicacao
<ul style="list-style-type: none">- nome : String- tecnologia : String- statusAtividade : boolean



2.2. DESCRIÇÃO DAS CLASSES

2.2.1. CLASSE ENDERECO

Endereco
- logradouro : String
- numero : int
- complemento : String
- bairro : String
- cidade : String
- estado : String

Descrição: Representa uma estrutura de dados reutilizável para armazenar informações de localização física. É utilizada por diversas outras classes para evitar a duplicação de atributos de endereço.

Atributos:

- - logradouro: String: Nome da rua ou avenida.
 - - numero: int: Número do imóvel.
 - - complemento: String: Informações adicionais (bloco, apartamento, etc.).
 - - bairro: String: Bairro da localização.
 - - cidade: String: Cidade da localização.
 - - estado: String: Estado (unidade federativa) da localização.
-

2.2.2. CLASSE PARCEIRO

Parceiro
- id : String
- nome : String
- contato : String
- endereco : Endereco

Descrição: Modela uma entidade ou empresa que colabora com a organização.

Atributos:

- - id: String: Identificador único do parceiro.

- - nome: String: Nome do parceiro.
- - contato: String: Informação de contato (telefone, e-mail).
- - endereco: Endereco: Objeto contendo o endereço do parceiro.

Relacionamentos: Possui uma associação de composição com a classe Endereco.

2.2.3. CLASSE DOADOR

Doador
<ul style="list-style-type: none"> - id : String - nome : String - contato : String - endereco : Endereco

Descrição: Representa uma pessoa física ou jurídica que realiza doações para a ONG.

Atributos:

- - id: String: Identificador único do doador.
- - nome: String: Nome do doador.
- - contato: String: Informação de contato.
- - endereco: Endereco: Objeto contendo o endereço do doador.

Relacionamentos: Possui uma associação com Endereco e é referenciada pela classe Doacao.

2.2.4. CLASSE PROGRAMA

Programa
<ul style="list-style-type: none"> - id : String - nome : String - descrição : String - campanhaVinculada : Campanha

Descrição: Modela uma iniciativa ou projeto contínuo da organização (*Dentistas do Bem* ou *Apolônias do Bem*), que pode englobar diversas campanhas.

Atributos:

- - id: String: Identificador único do programa.
- - nome: String: Nome do programa.
- - descricao: String: Descrição detalhada dos objetivos e atividades do programa.
- - campanhaVinculada: Campanha: Referência à campanha associada a este programa.

Relacionamentos: É referenciada pelas classes Voluntario, Paciente e Campanha.

2.2.5. CLASSE VOLUNTARIO

Voluntario
<ul style="list-style-type: none">- id : String- nome : String- profissao : String- contato : String- programasAtuacao : Programa- endereco : Endereco

Descrição: Representa um indivíduo que presta serviços voluntários para a organização, como um psicólogo ou um dentista.

Atributos:

- - id: String: Identificador único do voluntário.
- - nome: String: Nome do voluntário.
- - profissao: String: Profissão do voluntário (ex: Dentista, Advogado).
- - contato: String: Informação de contato.
- - programasAtuacao: Programa: Programa no qual o voluntário atua.
- - endereco: Endereco: Endereço do voluntário.

Relacionamentos: Possui associações com Endereco e Programa, e é referenciada pela classe Consulta.

2.2.6. CLASSE PACIENTE

Paciente
<ul style="list-style-type: none">- id : String- nome : String- idade : int- contato : String- endereco : Endereco- historico : String- programaVinculado : Programa

Descrição: Representa um beneficiário dos serviços da organização Turma do Bem.

Atributos:

- - id: String: Identificador único do paciente.
- - nome: String: Nome do paciente.
- - idade: int: Idade do paciente.
- - contato: String: Informação de contato.
- - endereco: Endereco: Endereço do paciente.
- - historico: String: Histórico ou prontuário do paciente.
- - programaVinculado: Programa: Programa ao qual o paciente está vinculado.

Relacionamentos: Possui associações com Endereco e Programa, e é referenciada pela classe Consulta.

2.2.7. CLASSE CAMPANHA

Campanha
<ul style="list-style-type: none">- id : String- nome : String- objetivo : String- metaFinanceira : double- programaVinculado : Programa

Descrição: Modela um evento ou esforço com início, meio e fim, geralmente para arrecadação de fundos ou para uma ação específica, vinculada a um programa maior.

Atributos:

- - id: String: Identificador único da campanha.

- - nome: String: Nome da campanha.
- - objetivo: String: Descrição do objetivo da campanha.
- - metaFinanceira: double: Valor financeiro que a campanha almeja arrecadar.
- - programaVinculado: Programa: Programa ao qual a campanha pertence.

Relacionamentos: Possui uma associação com Programa e é referenciada pela classe Doacao.

2.2.8. CLASSE DOACAO

Doacao
<ul style="list-style-type: none"> - id : String - valor : double - data : String - campanha : Campanha - doador : Doador

Descrição: Representa o ato de uma doação, registrando o valor, data e quem a realizou.

Atributos:

- - id: String: Identificador único da doação.
- - valor: double: Valor monetário doado.
- - data: String: Data em que a doação foi realizada.
- - campanha: Campanha: Campanha para a qual a doação foi destinada.
- - doador: Doador: Referência ao doador que realizou a doação.

Relacionamentos: Possui associações diretas com Campanha e Doador.

2.2.9. CLASSE CONSULTA

Consulta
<ul style="list-style-type: none">- id : String- data : String- requisitos : String- resultado : String- paciente : Paciente- dentista : Voluntario

Descrição: Modela um evento de atendimento, como uma consulta médica ou odontológica, conectando um paciente a um voluntário.

Atributos:

- - id: String: Identificador único da consulta.
- - data: String: Data da consulta.
- - requisitos: String: Requisitos ou preparações para a consulta.
- - resultado: String: Descrição do resultado ou encaminhamento da consulta.
- - paciente: Paciente: Referência ao paciente atendido.
- - dentista: Voluntario: Referência ao voluntário (neste contexto, um dentista) que realizou o atendimento.

Relacionamentos: Possui associações diretas com as classes Paciente e Voluntario.

2.2.10. CLASSE FINANCEIRO

Financeiro
<ul style="list-style-type: none">- id : String- anoFiscal : int- receitaTotal : double- despesas : double

Descrição: Classe agregadora para controle financeiro geral da organização em um determinado período.

Atributos:

- - anoFiscal: int: Ano de referência para o balanço financeiro.
 - - receitaTotal: double: Soma de todas as receitas (doações) no período.
 - - despesas: double: Soma de todas as despesas no período.
-

2.2.11. CLASSE CANALCOMUNICACAO

CanalComunicacao
<ul style="list-style-type: none">- nome : String- tecnologia : String- statusAtividade : boolean

Descrição: Modela os diferentes canais de comunicação que a organização utiliza para interagir com seus stakeholders.

Atributos:

- - nome: String: Nome do canal (ex: E-mail, WhatsApp, SMS).
- - tecnologia: String: Tecnologia ou plataforma utilizada.
- - statusAtividade: boolean: Indica se o canal está atualmente ativo (true) ou inativo (false).

O PRIVILÉGIO DE
PODER MUDAR VIDAS,

A COMPAIXÃO COMO
VALOR.

A TECNOLOGIA PARA
IMPULSIONAR

ISSO É
TURMA DO BEM,

ISSO É
TECH DO BEM.