**🩺 Resumo do Projeto: Sistema de Gestão de Consultas Médicas**

Este sistema permite:

* Cadastro de **médicos** e **pacientes**.
* **Agendamento** e **cancelamento** de consultas médicas.
* Geração de **relatórios em PDF** com os dados das consultas.
* Uso de **interface gráfica (GUI)** para interação do usuário.

**🧱 Estrutura e Função de Cada Classe**

**🔹 Consulta.java**

**Responsável por representar uma consulta médica agendada.**

* Contém atributos como data, hora, medico, paciente.
* Métodos getData(), getHora(), getMedico(), etc.
* Override de toString() para exibição formatada da consulta.

**🔹 ControladorConsulta.java**

**Classe central de controle do sistema.**

* Contém ArrayList para armazenar médicos, pacientes e consultas.
* Responsável por:
  + Adicionar médicos e pacientes.
  + Agendar e cancelar consultas.
  + Gerar relatórios (RelatorioPDF).
  + Validar dados usando Notificacoes.

**🔹 Entidade.java**

**Classe base para identificação única.**

* Pode conter ID, usado para herança por Pessoa.

**🔹 ICadastravel.java**

**Interface usada para garantir que classes tenham um getCpf().**

* Implementada por Paciente e Medico.

**🔹 Medico.java**

**Classe que representa um médico.**

* Herda de Pessoa, implementa ICadastravel.
* Tem especialidade médica como atributo.
* Métodos de acesso e sobrescrita de tipo (getTipo()).

**🔹 Paciente.java**

**Classe que representa um paciente.**

* Herda de Pessoa, implementa ICadastravel.
* Contém CPF, nome, e métodos relacionados.

**🔹 Pessoa.java**

**Classe abstrata para dados comuns entre médico e paciente.**

* Tem nome, CPF e métodos de acesso.
* Usada como base por Medico e Paciente.

**🔹 Notificacoes.java**

**Classe utilitária para validações e mensagens.**

* Valida formatos de:
  + CPF
  + Data (dd/MM/yyyy)
  + Hora (HH:mm)
* Usa exceções para tratamento de erros.

**🔹 RelatorioPDF.java**

**Gera o relatório de consultas em PDF.**

* Usa a biblioteca iText 2.1.7.
* Gera arquivo com lista de consultas agendadas.

**🔹 ProgramaGUI.java**

**Classe principal com main().**

* Instancia ControladorConsulta.
* Inicializa a interface gráfica chamando TelaDeEntradaDoPG.

**🔹 TelaDeEntradaDoPG.java**

**Interface gráfica com Swing.**

* Cria janelas, botões, campos de entrada.
* Permite:
  + Cadastrar médicos/pacientes.
  + Agendar/cancelar consultas.
  + Gerar relatório.
* Interage diretamente com ControladorConsulta.

**✅ Como os Requisitos São Atendidos no Funcionamento**

| **Requisito** | **Como é implementado no código** |
| --- | --- |
| **Herança** | Medico e Paciente → Pessoa → Entidade |
| **Polimorfismo** | getTipo() sobrescrito; uso genérico de Pessoa |
| **Coleções** | ArrayList<Consulta> no controlador |
| **Classe Abstrata** | Pessoa |
| **Interface** | ICadastravel |
| **Camadas** | Separação entre modelo, controle e interface gráfica |
| **GUI** | Swing com TelaDeEntradaDoPG |
| **Exceções** | try-catch no PDF e nas validações de Notificacoes |