

App de Gestão de Clínica Médica

Lucas Martins Vasconcelos - 12111BCC056

Millena Gená Pereira - 12111BCC055

Rafael Guimarães - 12021BCC002

Yves Yan Esteves Pereira - 12021BCC022

Para o desenvolvimento da aplicação de gestão de clínica médica, serão criadas as seguintes classes: **Endereço**, **Clínica**, **Médico**, **Pessoa**, **Medicamento**, **Agenda** e **RegistroConsulta**. Sendo que:

1) Endereço

A classe **Endereco** será criada para armazenar detalhes a respeito do objeto endereço que será criado nas classes **Clinica** e **Pessoa**. Ela possuirá os seguintes atributos:

- logradouro: string
- cep: string
- numero: int
- bairro: string
- cidade: String
- estado: String

Os métodos dessa classe serão, caso seja necessário, os gets e sets que fizerem sentido no momento de desenvolvimento da aplicação.

2) Agenda

A classe **Agenda** será responsável por armazenar os horários disponíveis para atendimento de cada médico:

- dia: Int
- mes: Int
- ano: Int
- hora: String
- status: Boolean

Os métodos dessa classe serão os gets e sets que fizerem sentido no momento de desenvolvimento da aplicação. Todavia, um método **get** que com certeza será necessário é o **getStatus** para consulta da disponibilidade de atendimento em determinada data e horário.

3) Pessoa

A classe **Pessoa** será a classe mãe das classes **Medico** e **Paciente** e possuirá os seguintes atributos:

- nome: String
- cpf: String
- idade: Int
- rg: String
- nomeMae: String
- telefone: Int
- email: String
- endereco: Endereco

Os métodos dessa classe serão, caso seja necessário, os gets e sets que fizerem sentido no momento de desenvolvimento da aplicação.

4) Medico

A classe **Medico** será uma das classes filha da classe **Pessoa** e possuirá os seguintes atributos:

- especialidade: String
- duracaoConsulta: String
- listaConvenios: ArrayList<String>
- registroConsultas: ArrayList<RegistroConsulta>
- agenda: ArrayList<Agenda>

Além disso, ela possuirá os métodos:

- **getAgenda**: para o médico conseguir consultar sua agenda e horários por meio do atributo agenda
- **getHistPaciente**: esse método recebe como argumento um período determinado de tempo e o nome de um paciente. Como resposta, ele retorna o histórico de consultas desse paciente com o respectivo médico (no período de tempo determinado) por meio do atributo/objeto registroConsultas
- **getHistConsultas**: esse método recebe como argumento um período determinado de tempo e assim como o método anterior, este também irá acessar o atributo/objeto registroConsultas, no entanto apenas dados como a data da consulta e o nome do paciente serão apresentados

5) Paciente

A classe **Paciente** será uma das classes filha da classe **Pessoa** e possuirá os seguintes atributos:

→ historicoDoencas: ArrayList<String>

Além disso, ela possuirá os métodos:

→ **getHistoricoDoenca**: para que seja possível acessar o histórico de doenças do paciente

6) Medicamento

A classe **Medicamento** possuirá os seguintes atributos:

→ nome:String
→ faixa: String
→ quantidade: int
→ tipo: String

Além disso, ela possuirá os métodos:

→ **getHistoricoUtilizacao**: para que seja possível acessar o histórico de utilização dos medicamentos disponíveis na clínica

7) RegistraConsulta

A classe **RegistraConsulta** será criada com o intuito de armazenar os detalhes referentes a cada consulta realizada. Ela será utilizada pelo médico para registrar os sinais vitais do paciente além da data da consulta e medicamentos receitados. Sendo assim, possuirá os seguintes atributos:

→ nomePaciente: String
→ nomeMedico: String
→ especialidade: String
→ dataConsulta: datetime
→ bpm: String
→ pressao: String;
→ temperatura: String
→ diagnostico: String
→ medicamentos: ArrayList<Medicamento>

→ obs: String

Os métodos dessa classe serão os gets e sets que fizerem sentido no momento de desenvolvimento da aplicação.

8) Clínica

A classe **Clínica** será criada com o intuito de armazenar os detalhes referentes a cada clínica existente. Ela será responsável por cadastrar médicos, pacientes e também fornecer os relatórios solicitados. Sendo assim, possuirá os seguintes atributos:

- nome: String
- endereco: Endereco;
- medicos: ArrayList<Medico>
- pacientes: ArrayList<Paciente>
- medicamentos: ArrayList<Medicamento>
- registroConsultas: ArrayList<RegistroConsulta>

Além disso, ela possuirá os métodos:

- **cadastrarMedico**: para cadastrar objetos da classe Medico que trabalham na clínica em questão
- **cadastrarPaciente**: para cadastrar objetos da classe Paciente que são atendidos na clínica em questão
- **agendaHorario**: para agendar horários de pacientes com os médicos, de acordo com a agenda disponibilizada por cada médico
- **relatorioHistPaciente**: esse método vai receber o nome do paciente e uma faixa de data como argumentos e será responsável por consultar o objeto *registraConsulta* dentro de cada um dos objetos da lista *medicos*. Ou seja, esse método vai retornar um relatório com todas as consultas de um paciente com todos os médicos em um determinado período de tempo.
- **relatorioConsultasMedico**: esse método vai receber uma faixa de datas como argumento. Assim, será responsável por consultar o objeto *registraConsulta* dentro de cada um dos objetos da lista *medicos*. Assim será retornado um relatório com o histórico de todas as consultas de todos médicos na faixa de tempo informada