

Trabalho Prático 5 de PM

Sistema de Agendamento

Douglas Rodrigues de Almeida

douglasralmeida@live.com

1. Introdução

O objetivo do trabalho foi a criação de um programa modular para a gestão de agendamento de atendimentos de uma clínica médica.

2. Implementação

Foi implementada uma classe Pessoa para representar pessoas cadastradas no sistema. Essa classe possui subclasses, Atendente e Cliente, para distinguir os clientes dos atendentes da clínica. Os atendentes ainda possui subclasses para distinguir o tipo de atendente: Medico e Tecnico.

O agendamento é representado pela classe Atendimento. Os atendimentos do dia estão agrupados na classe Data, que representa um dia de atendimento. Possui sua vez, a classe Data está agrupada na classe Agenda, que representa a agenda de um atendente no mês.

Os serviços da clínica possui uma interface Servico. Dela descende as classes Exame e Especialidade. A classe Exame representa os serviços oferecidos pelos técnicos. A classe Especialidade representa os serviços oferecidos pelos médicos. Essas duas classes estão agrupadas na classe Servicos que armazena todos os serviços da clínica.

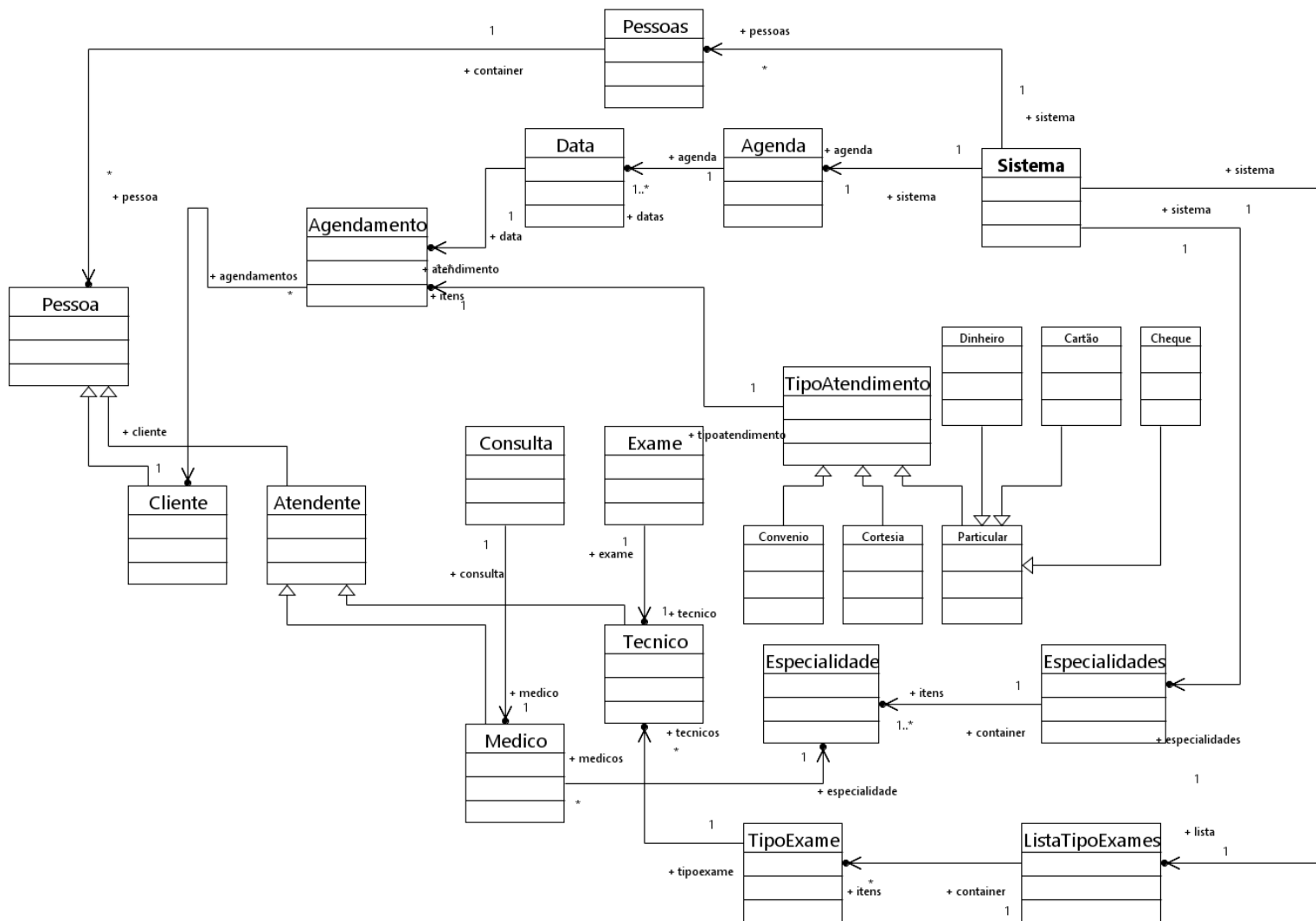
O pagamento do atendimento está representado pela classe Pagamento. Dela descende as classes Convenio, Cortesia, Dinheiro, Cartao e Cheque. Todas as classes estão agrupadas na classe Pagamentos. Cada classe controla a autorização para o agendamento.

Todos os objetos do sistema estão armazenados na classe Aplicacao, inclusive as classes que representam as janelas. Também foram implementadas a classe Validadores para validar dados em formulários, Buscador para fazer pesquisas, Agendador para fazer o agendamento e ArgumentoInvalidoException e AgendamentoException que são usadas para tratamento de exceções geradas quando as informações digitadas pelo usuário estão inválidas e durante o agendamento. Por fim foi criada a interface AgendamentoListener que controla o disparo ocorrido quando um agendamento é concluído e dois componentes: AppPainel e AppButton. O primeiro componente agrupa componentes gráficos para a geração da interface e o segundo adiciona um container genérico em botões para auxiliar na geração do calendário para escolher a data do agendamento.

Foi implementada uma classe para testes unitários através do JUnit chamada PessoaTest que verifica se a manipulação de dados da classe Pessoa está sendo realizada corretamente.

Na página seguinte está o diagrama de classes.

3. Diagrama de classes



4. Interface Gráfica

A interface gráfica da aplicação é representada pela classe `JanelaPrincipal`. Existem duas janelas representadas pelas classes `JanelaFuncionario` e `JanelaServico`.

Na lateral esquerda, o usuário escolhe uma função do programa. Foram implementadas o cadastro de funcionários e serviços. Foi implementada parcialmente a função de agendamento.

Para testar a interface gráfica, crie um serviço e, em seguida, um funcionário alocado no serviço criado. Em seguida faça o agendamento.

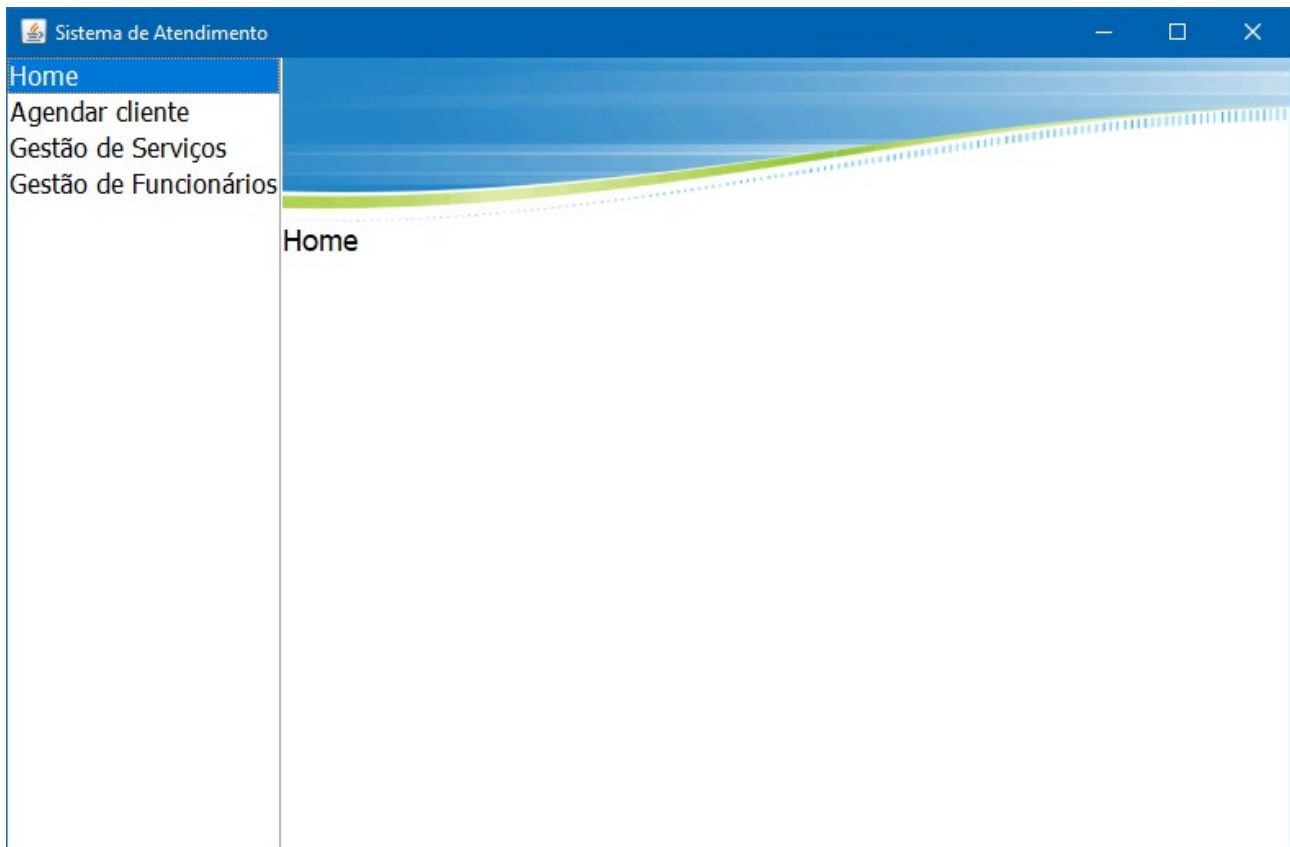


Foto da tela inicial

Sistema de Atendimento

- Home
- Agendar cliente**
- Agenda do Dia
- Gestão de Serviços
- Gestão de Funcionários
- Configurações

Agendar cliente

CPF do cliente:

Cliente: DOUGLAS RODRIGUES

Tipo de Serviço: ☒ Consulta Médica ☐ Exame Médico

Serviço:

	DOM	SEG	TER	QUA	QUI	SEX	SAB
						1	2
	3	4	5	6	7	8	9
Data:	10	11	12	13	14	15	16
	17	18	19	20	21	22	23
	24	25	26	27	28	29	30
	31						

Horários disponíveis:

Foto do Agendamento parcialmente implementado

Funcionário

NOVO - FUNCIONÁRIO

Novo funcionário

Dados Gerais

Nome:

CPF:

Matrícula:

Tipo: ☐ Técnico ☒ Médico

Serviço:

Foto do cadastro de funcionários