

1. Introdução

O sistema de gerenciamento de clínica médica foi desenvolvido para administrar o agendamento de consultas, o cadastro e histórico de pacientes e médicos, a prescrição de exames e medicamentos, e a gestão de pagamentos. O projeto foi estruturado para separar claramente os dados do domínio, a lógica de negócio e a interface de interação com o usuário, permitindo uma aplicação modular, flexível e de fácil manutenção.

2. Associações

Entre os Elementos do Domínio

- **Paciente e Consulta:** Cada paciente mantém um histórico de consultas, o que representa uma associação de “um para muitos”. Essa relação possibilita o acompanhamento dos atendimentos realizados e permite a verificação de agendamentos para evitar conflitos.
- **Médico e Consulta:** Um médico pode atender a várias consultas, sendo sua agenda composta por múltiplos registros de atendimento. Essa associação é fundamental para verificar a disponibilidade do profissional antes de agendar uma nova consulta.
- **Consulta com Exame/Medicamento:** Durante uma consulta, podem ser prescritos diversos exames e medicamentos. Dessa forma, uma consulta pode estar associada a vários exames ou registros de prescrição, refletindo as práticas clínicas de solicitar procedimentos ou tratamentos adicionais.
- **Consulta e Pagamento:** Cada consulta (ou os serviços a ela relacionados, como exames) possui um pagamento vinculado, o que garante que o serviço prestado seja cobrado corretamente. Essa associação facilita o controle financeiro e a verificação de pendências.

Entre os Módulos de Serviço e as Entidades

- Os módulos de serviço foram desenvolvidos para executar operações essenciais (como cadastro, busca, atualização e remoção) sobre os registros do sistema. Eles operam diretamente sobre as entidades do domínio e, por meio de uma estrutura genérica, permitem aplicar a mesma lógica de negócio para diferentes tipos de registros, evitando a duplicação de código.

Entre os Módulos de Interface e os Serviços

- Os componentes de interface (como os menus) interagem com o usuário e delegam as operações à camada de serviços. Essa associação garante que a interface seja responsável apenas pela coleta de dados e pela exibição de informações, enquanto a lógica de negócio é mantida separada nos serviços.

3. Herança

Utilização da Herança para Centralizar Comportamentos Comuns

- **Entidade Base de Pessoa:** Para evitar a repetição de atributos comuns, foi criada uma classe base que reúne informações essenciais, como nome, CPF e data de nascimento. Essa entidade serve como base para os registros de paciente e médico, que herdam essas informações e comportamentos comuns, e adicionam suas características específicas (como o histórico de consultas para pacientes e o CRM e especialidade para médicos).
- **Serviços Base:** A camada de serviços também utiliza herança, através de uma classe base genérica que define operações fundamentais. Cada serviço específico – seja para pacientes, médicos, consultas, exames ou pagamentos – estende essa classe base, garantindo uma estrutura uniforme e facilitando a manutenção.

4. Polimorfismo

Polimorfismo por Sobrescrita

- No sistema, métodos definidos na classe base são reimplementados de forma específica nas subclasses. Por exemplo, a entidade base de pessoa possui um método para exibir informações, que é adaptado de forma diferente tanto para pacientes quanto para médicos. Isso permite que, ao chamar esse método em um objeto tratado como do tipo base, a implementação correta e específica para o objeto seja executada.

Polimorfismo por Sobrecarga

- Em diversos pontos da lógica de negócio, métodos com o mesmo nome foram implementados com assinaturas diferentes. Essa sobrecarga possibilita que a mesma operação (como agendar uma consulta) seja realizada com conjuntos distintos de informações, adaptando-se aos diferentes cenários de uso sem a necessidade de criar métodos com nomes distintos.

Polimorfismo Paramétrico

- A utilização de uma estrutura genérica na camada de serviços permite que operações comuns sejam aplicadas a diferentes tipos de registros de forma segura e reutilizável. Essa parametrização assegura que as operações realizadas em pacientes, médicos, consultas, exames e pagamentos sejam específicas para cada tipo, evitando conversões desnecessárias e promovendo a consistência do código.

5. Papel de Cada Classe no Projeto

- **Entidade Base de Pessoa:** Centraliza atributos e comportamentos comuns a todas as pessoas, servindo de base para as classes de paciente e médico.
- **Paciente:** Representa os registros dos pacientes e gerencia seu histórico de consultas, permitindo o acompanhamento dos atendimentos e a verificação de pendências.

- **Médico:** Armazena informações dos médicos, como CRM e especialidade, e gerencia sua agenda de consultas para verificar disponibilidade antes de novos agendamentos.
- **Consulta:** Conecta pacientes e médicos em atendimentos, registrando detalhes como data, horário, status e, quando aplicável, os exames, prescrições e pagamentos associados.
- **Exame e Medicamento/Prescrição:** Registam os procedimentos e tratamentos prescritos durante uma consulta, facilitando o acompanhamento e a rastreabilidade dos serviços prestados.
- **Pagamento:** Gerencia as informações financeiras dos atendimentos, vinculando cada cobrança ao respectivo serviço prestado e permitindo o controle de pendências.
- **Serviço Base e suas Especializações:** Cada módulo de serviço é responsável por operar sobre um tipo específico de registro, utilizando uma estrutura genérica para aplicar operações fundamentais de cadastro, busca, atualização e remoção de forma consistente.
- **Menus e Interfaces de Usuário:** Responsáveis pela interação com o usuário, essas classes delegam as operações de negócio aos serviços correspondentes, garantindo uma clara separação entre a apresentação e a lógica de processamento.