Instituto Federal de Santa Catarina - Câmpus Gaspar
Análise e Desenvolvimento de Sistemas
Padrões de Projeto de Software
Sarah Carolina de Camargo

Padrões de Projeto Utilizados no Sistema Hospitalar

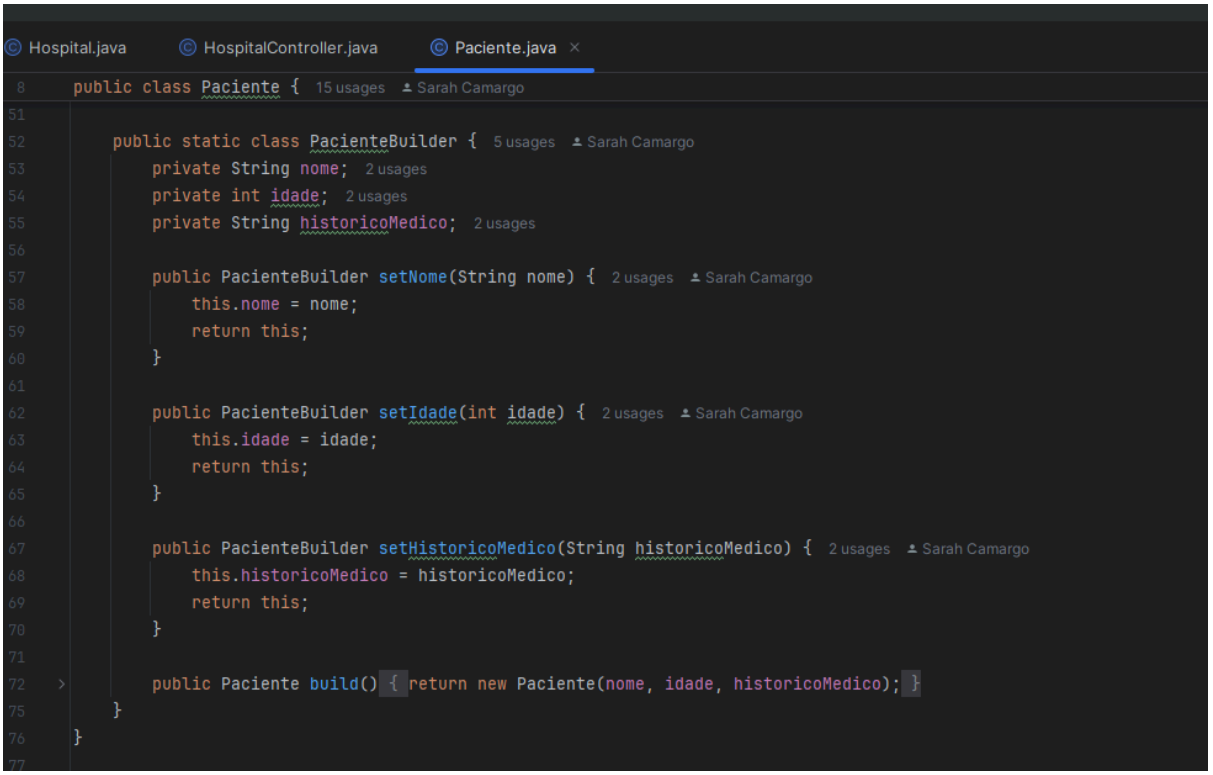
1. Singleton

O padrão Singleton foi usado na classe “Hospital” para garantir que haja apenas uma única instância do hospital no sistema. Isso é útil porque o hospital precisa centralizar o gerenciamento de pacientes, médicos e consultas. Com o Singleton, temos certeza de que sempre estamos trabalhando com a mesma instância do hospital, evitando inconsistências.
Implementa-se com um construtor privado e um método estático que retorna a única instância da classe.

```
public class HospitalController {  1 usage  Sarah Camargo

    private static Hospital hospitalInstance;  3 usages

    private HospitalController() {}  no usages  Sarah Camargo

    public static Hospital getInstance() {  1 usage  Sarah Camargo
        if (hospitalInstance == null) {
            hospitalInstance = new Hospital();
        }
        return hospitalInstance;
    }
}
```

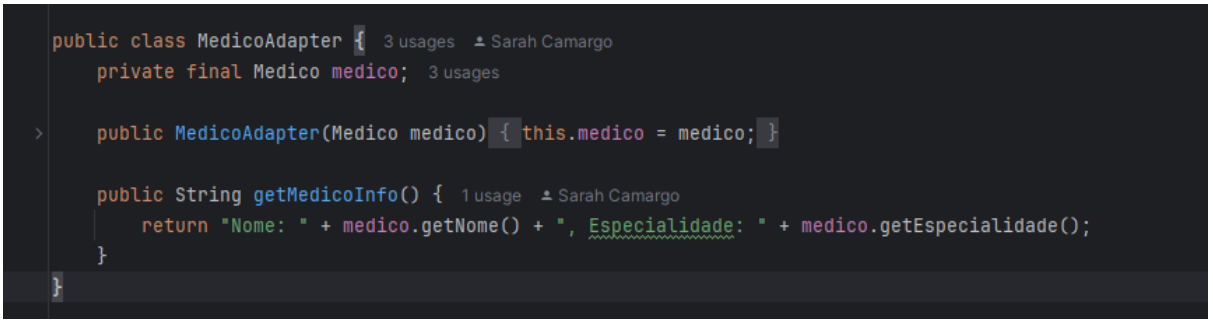
2. Builder

O padrão Builder foi aplicado na classe “Paciente” para facilitar a criação de objetos que têm muitos atributos. Em vez de usar vários construtores, o Builder permite que você monte o objeto passo a passo de maneira mais clara. Isso torna o código mais legível e fácil de modificar, especialmente quando nem todos os atributos precisam ser definidos na criação do objeto.
Implementa-se como uma classe interna estática dentro da classe “Paciente”, permitindo a configuração de atributos de forma fluente e intuitiva.



3. Adapter (Estrutural)

O padrão Adapter foi usado na classe “MedicoAdapter” para permitir que a classe “Medico” seja compatível com outros sistemas que precisam de uma interface diferente. O Adapter age como um tradutor, permitindo que “Medico” seja usado onde uma interface específica é necessária, sem modificar o código original da classe.



4. Observer (Comportamental)

O padrão Observer foi aplicado na classe “ConsultaManager” para gerenciar notificações quando uma consulta é agendada ou cancelada. Com esse padrão, diferentes partes do sistema podem reagir automaticamente a esses eventos, como enviar um email ou SMS ao paciente. O Observer facilita a adição de novos tipos de notificações no futuro, sem precisar alterar o código existente.