



Descrição da Arquitetura do Sistema

A arquitetura descrita no diagrama segue o padrão de **Arquitetura em Camadas (Layers)**, promovendo uma estrita separação de responsabilidades entre a interface com o usuário, a lógica de negócio e o armazenamento de dados. O sistema é modular e utiliza interfaces formalmente definidas para desacoplar a camada de apresentação da camada de serviço, garantindo flexibilidade e facilidade de manutenção.

Abaixo, segue uma descrição detalhada de cada componente e camada do sistema:

1. Camada de Apresentação (<<layer>>)

É a camada superior responsável pela interação direta com o usuário (Gerente). Suas principais responsabilidades incluem a coleta de dados de entrada, a validação de formatos e a exibição de resultados.

- Esta camada não possui lógica de negócio complexa.
- Ela atua como uma "consumidora" dos serviços, dependendo estritamente das **Interfaces de Serviço** para realizar qualquer operação no sistema.

2. Interfaces de Serviço (<<interface>>)

Constituem o "contrato" entre a apresentação e o serviço. São classes abstratas puras que definem *o que* o sistema pode fazer, sem expor *como* é feito. Isso garante o desacoplamento, permitindo que a implementação do serviço mude sem afetar a apresentação.

- **IAuthService**: Define operações de autenticação e gestão da conta do gerente.
- **IHotelService**: Define operações de CRUD (Create, Read, Update, Delete) para hotéis.
- **IQuartoService**: Define operações de gestão de quartos vinculados a hotéis.
- **IHospedeService**: Define operações de gestão de hóspedes.
- **IReservaService**: Define operações de criação e gestão de reservas.

3. Camada de Serviço (<<layer>>)

É o núcleo da lógica de negócio do sistema. Esta camada implementa as **Interfaces de Serviço**, processando as regras de negócio e garantindo a integridade das transações.

- Ela recebe requisições da camada de apresentação através das interfaces.
- Ela se comunica com a camada de armazenamento para persistir ou recuperar informações.
- Responsável por garantir que regras como "não pode haver conflito de datas em reservas" sejam cumpridas.

4. Camada de Armazenamento de Dados (<<component>>)

Representa os módulos responsáveis pela persistência dos dados do sistema. A camada de serviço depende destes módulos para armazenar o estado das entidades.

- **Gerentes:** Armazena dados de login e perfil dos administradores.
- **Hospedes:** Armazena informações pessoais dos clientes.
- **Hoteis:** Armazena os dados cadastrais dos estabelecimentos.
- **Quartos:** Armazena as unidades habitacionais e suas características.
- **Reservas:** Armazena os registros de locação, vinculando hóspedes a quartos.

Fluxo de Dados

1. O usuário solicita uma operação na **Camada de Apresentação** (ex: "Cadastrar Hotel").
2. A Apresentação valida os dados básicos e chama o método correspondente na interface IHotelService.
3. A **Camada de Serviço**, que implementa essa interface, recebe os dados.
4. O Serviço aplica as regras de negócio necessárias.
5. O Serviço envia os dados processados para o módulo de armazenamento correspondente (Hoteis).
6. O resultado da operação (sucesso ou erro) é retornado para a Apresentação exibir ao usuário.

Conclusão

A arquitetura projetada atende aos requisitos de modularidade e desacoplamento exigidos. O uso de interfaces como mediadoras entre as camadas facilita o desenvolvimento paralelo pela equipe e simplifica a realização de testes, permitindo uma evolução sustentável do software de gestão hoteleira.