

# Seu Cantinho - Relatório

Heloísa Dias Viotto

Guilherme Eduardo G. da Silva

27 de outubro de 2025

## 1 Introdução

Este documento tem como finalidade ser um relatório para o trabalho da disciplina Design de Software lecionada pela Professora Simone Dominico.

## 2 Estilo Arquitetural

O estilo arquitetural escolhido foi o de microsserviços, devido a sua vantagens para o trabalho.

A vantagem que mais se destaca é a escalabilidade, solicitado pelo enunciado do trabalho. Além disso, essa arquitetura se destaca por ser tolerante a falhas, ou seja, caso um modulo apresente qualquer tipo de problema, não prejudicará a aplicação como todo, bem como facilita a manutenção em pontos específicos.

O microsserviço cria vários serviços independentes uns dos outros, cada um é responsável por uma função, sendo a comunicação realizada através de trocas de mensagens. Para auxiliar nessa tarefa, foi definido o padrão de comunicação por meio de outro estilo arquitetural, denominado “REST”, o qual permite uma comunicação via interfaces (HTTP) entre os demais módulos da aplicação.

Embora a arquitetura microsserviço seja complexa e de difícil implementação, esse problema seria compensado com um front-end/interface web simplificada, por conta da falta de familiaridade da Dona Maria com tecnologias e afins.

## 3 Detalhes de Implementação

Visando uma solução mais robusta do “Seu Cantinho”, foram definidos algumas implementações que

mantêm o sistema coerente, evitando problemas de condições de corrida entre espaços para serem reservados, perda de informações e um baixo tempo nas operações do sistema.

Nossa estratégia para resolver as condições de corrida seriam aplicar uma teoria baseada nas primitivas *lock* e *unlock*, o qual após o cliente confirmar o local para ser reservado, esse espaço seria bloqueado até a finalização da reserva (pagamento ou sinal) ou após um limite de tempo estabelecido.

Pensando em manter a consistência e a independência dos dados, cada módulo da aplicação possui seu próprio banco de dados, garantindo isolamento e evitando acoplamentos indesejados entre os serviços. Para intermediar o acesso a esses dados, foi adotado o padrão de projeto *Repository*, responsável por encapsular as operações de persistência e abstrair a comunicação entre o serviço e o banco de dados.

Pelo fato do estilo arquitetural selecionado ser o microsserviço, podemos utilizar de vários servidores (Containers - Docker) separados para cada módulo, dessa forma se torna menos dependente, fazendo com que seja possível processar várias requisições simultâneas, reduzindo o tempo de resposta do sistema e, consequentemente, realizando opções mais rápidas.

## 4 Diagramas UML

A seguir, encontram-se os diagramas UML de classe e componentes, sendo, respectivamente, as Figuras 1 e 2.

Figura 1: Diagrama UML de Classe.

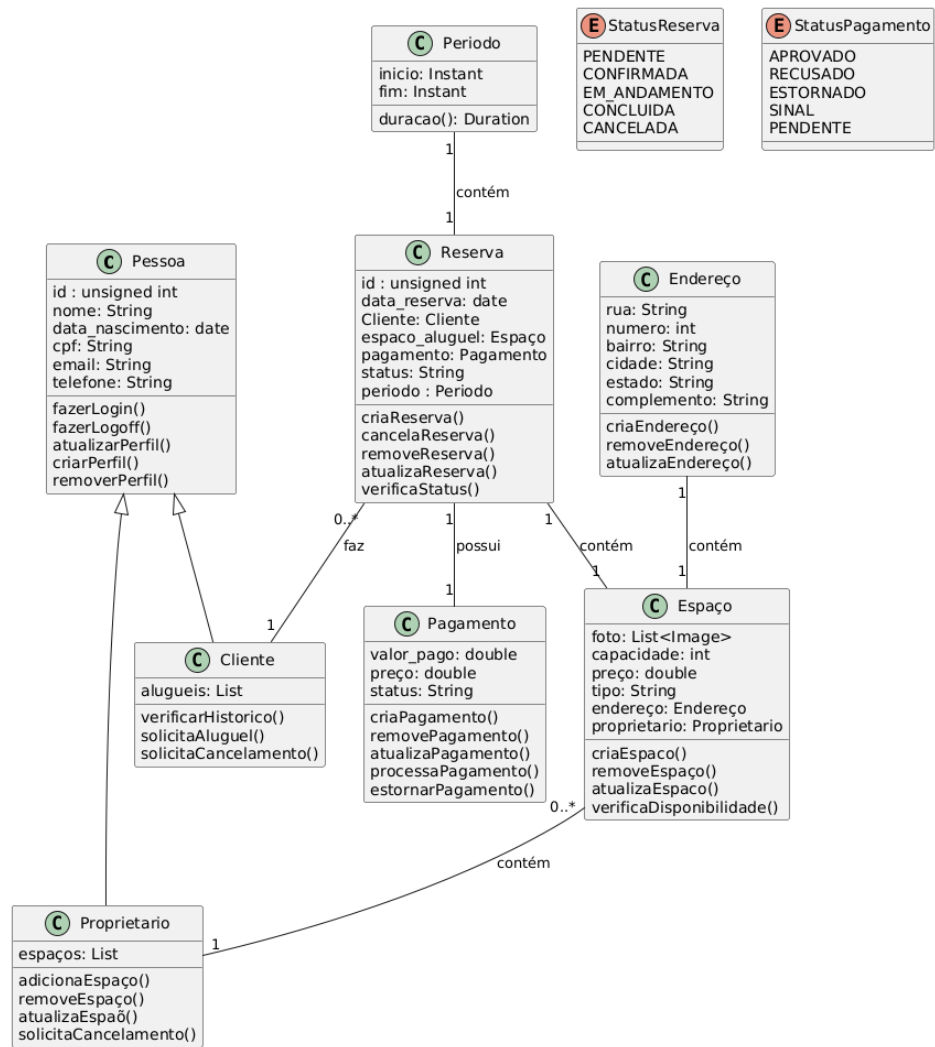


Figura 2: Diagrama UML de Componentes.

