

# Análise da Arquitetura do Projeto Refatorado

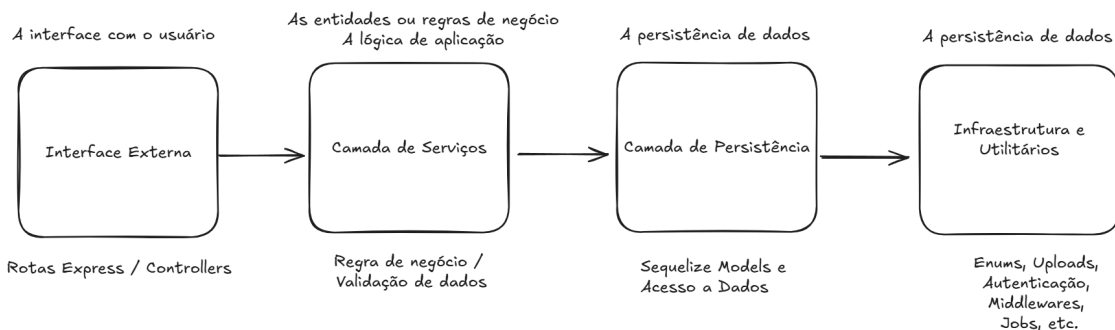
## Alunos

Bruno José Dimon Boger  
Douglas Kuerten

## Identificação da Arquitetura

A arquitetura utilizada no projeto refatorado é baseada no padrão de Arquitetura em Camadas, com elementos da Clean Architecture. As responsabilidades estão separadas em módulos bem definidos, promovendo a manutenção e evolução do sistema. As principais camadas são: Interface: controladores Express (Controllers) Aplicação / Negócio: Serviços (Services) Persistência de dados: Models Sequelize Infraestrutura: Middleware, Uploads, Autenticação, Jobs, Enums, Utils

## Diagrama da Arquitetura



## Análise dos Pontos

### O projeto respeita a separação de responsabilidades?

Sim. Cada camada tem uma responsabilidade clara e bem delimitada, facilitando a manutenção e entendimento do sistema.

### É fácil trocar a interface (ex: trocar um framework frontend)?

Sim. Como o backend expõe uma API REST, ele é independente da interface. Um novo frontend pode ser integrado facilmente.

**Os módulos têm baixo acoplamento e alta coesão?**

Sim. Cada serviço e módulo encapsula sua própria responsabilidade, com dependências bem definidas.

**Existe alguma dependência invertida?**

Parcialmente. A lógica de negócio ainda depende diretamente do Sequelize. Uma refatoração para interfaces (contracts) e injeção de dependências tornaria a arquitetura mais limpa.

**Algo que você mudaria na arquitetura?**

Criar camada de Repositórios para isolar o Sequelize  
Extrair regras para entidades de domínio  
Aplicar inversão de dependência completa  
Adicionar testes unitários nos serviços  
Criar DTOs para transferência de dados entre camadas