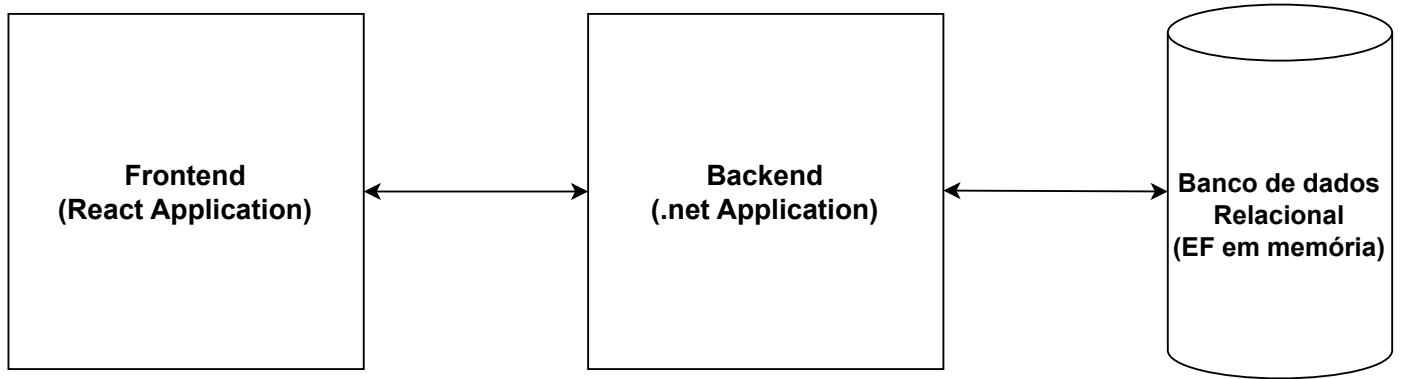
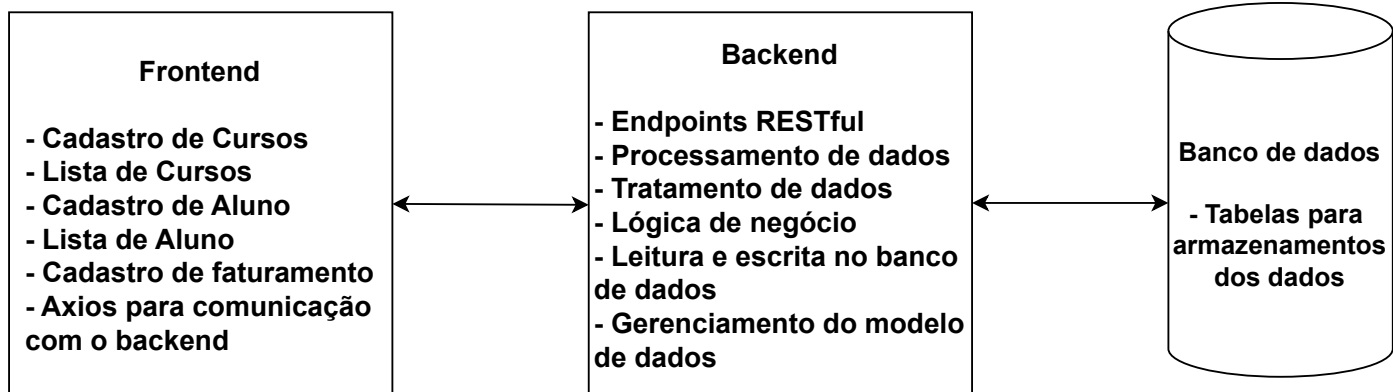


## Visão Macro



1. **Frontend:** Interage com o usuário e faz requisições ao backend;
2. **Backend:** Processa as requisições recebidas do frontend, interage com o banco de dados e fornece respostas;
3. **Banco de dados:** Armazena os dados da aplicação

## Visão Detalhada



## Frontend

- Aplicação criada utilizando o framework ReactJS;
- Responsabilidade somente de interagir com o usuário através de suas telas;
- Não se comunica diretamente com o banco de dados
- Todo o acesso à dados é feito através da aplicação de backend utilizando protocolo http
- Rotas para as telas:
  - Cadastro de curso (/course)
  - Listagem de cursos (/list-courses)

## Backend

- Aplicação criada utilizando o framework .net core e linguagem c#;
- Arquitetura de micro serviços, separando as responsabilidades em camadas;
- **Web:** Camada responsável na disponibilização dos endpoints para leitura e escrita dos dados usando o padrão RESTful, onde cada endpoint é responsável somente por uma entidade do domínio;
- **Application:** Camada responsável em centralizar e processar as regras de negócio, recebe comunicação da camada **Web** através de contratos (interface) previamente definidos;
- **Repository:** Camada responsável em receber comunicação da camada **Application** através de contratos previamente definidos, essa camada possui somente interfaces, provendo as operações necessárias para acesso aos dados;
- **Persistence:** Camada que possui as implementações da camada **Repository**, nessa camada é onde está implementada a configuração do Entity Framework, mas poderia ser qualquer outro ORM e consequentemente qualquer outro banco de dados;
- **Domain:** Camada que possui o modelo de domínio da aplicação, essa camada é composta por classes anêmicas, ou seja, possui somente propriedades e métodos relativos à acesso às propriedades;
- **DependencyInjection:** Camada responsável em centralizar a injeção de dependência dos objetos criados, não exerce qualquer função do ponto de vista funcional, sendo somente uma forma de organização melhor do projeto;

## Banco de dados

- Banco de dados em memória, ou seja, os dados são persistidos somente durante a execução da aplicação;