

# Validação de APIs

# Validação de APIs

- Uma das maiores dores de cabeça de um programador é tratar dados inconsistentes no banco de dados
- É confiar que dados como a seguir vão vir corretos:
  - CEP
  - Nome da Cidade
  - CPF
  - Email
  - Telefone
- Prevenir é muito mais barato do que gastar muito tempo apagando incêndio por falta de validação na API

# Validação de APIs

- A estrutura ideal de validação em um sistema
  - Validação no cliente (front-end, como app, web, ou desktop)
  - Validação na entrada da API
  - Validação a nível de domínio

# Até a próxima aula!

# O que é **Fluent Validation**

# O que é Fluent Validation

- Biblioteca utilizada para validação de objetos, portanto permitindo ser utilizada para verificar nossos modelos de entrada (Commands / Input Models)
- Instalada na aplicação através do pacote **FluentValidation.AspNetCore**
- São criadas classes de validação para cada modelo de entrada a ser validado

# O que é Fluent Validation

- Para utilizar esta biblioteca, o fluxo é:
  - Configurar o Fluent Validation de forma global
  - Criar uma classe que herde de **AbstractValidator<T>**
  - Implementar as regras no construtor através do método **RuleFor**
- Caso prefira fazer de forma manual, é possível utilizar injeção de dependência para obter o objeto **IValidator<T>** , onde terá acesso ao método Validate e seu resultado, checando a propriedade IsValid

# O que é Fluent Validation

- Funcionalidades do Fluent Validation
  - Validações de propriedades simples
  - Validações de propriedades complexas
  - Validação de listas
  - Múltiplos validadores para a mesma classe

# O que é Fluent Validation

- Métodos de validação mais utilizados
  - NotNull, NotEmpty
  - Equal, NotEqual
  - Length, MaxLength, MinLength
  - LessThan, LessThanOrEqualTo, GreaterThan, GreaterThanOrEqualTo
  - Must
  - EmailAddress
  - CreditCard
  - IsInEnum

# Até a próxima aula!