# Camada Domain

Modelos de domínio, regras de negócio e interfaces;

**Não Possui nenhuma dependência.**

Modificador “SEALED” nas classes de domínio -> Isolando a classe de domínio, de forma que ela não possa ser herdada.

## Estrutura de Pastas

A estrutura de pastas do Projeto Domain são: Entidades, Enums, Interfaces, Validacao.

* Entidades: Classes de domínio (Cliente, Usuario, Operador etc.)
* Interfaces: Interfaces Repository para manipulação dos domínios
* Validacao: Classes para validação dos domínios
* Enums: Enumerados para utilização nas classes de domínio

### Classe DomainExceptionValidation

Classe criada para validação das classes de domínio. Herda de ***Exception.*** Nesta classe, o construtor recebe um erro, que é repassado para a classe base *Excetion.*

O Método ***When*** recebe se há erros e, caso verdadeiro, lança uma exceção com a mensagem enviada.

# Camada Infra.Data

Entity Framework Core (EF Core), Contexto, Configurações, Migrations e Repository

**Possui dependência com o Projeto: Domain**

## Estrutura de Pastas

A estrutura de pastas do Projeto Domain são: Context, EntitiesConfiguration, Migrations, Repositories.

* Context: Classe de Contexto para ser utilizada pelo EF Core, que define as tabelas a serem criadas e onde localizar suas configurações
* EntitiesConfiguration: Configurações das classes de domínio para criação das tabelas no banco de dados pelo EF Core
* Migrations: Classes geradas usando o Migration para criação do banco de dados.
* Repositories: Repositories que implementam as interfaces Repository da camada Domain. Acessam e manipulam os dados no banco de dados.

# Camada Infra.IoC

Dependency Injection, registros de serviços e tempo de vida.

**Possui dependência com os Projetos: Domain, Application, Infra.Data**

# Camada Application

Regras de domínio da Aplicação, mapeamentos, serviços, DTOs e CQRS

**Possui dependência com o Projeto: Domain**

# Camada WebAPI

Todas as APIs e Controllers

**Possui dependência com o Projeto: Infra.IoC**