



PÓS-GRADUAÇÃO MIT ENGENHARIA DE SOFTWARE COM .NET

Projeto de Bloco: Engenharia de Software e Modelagem

Integrantes:

ALUNO: ALEXANDER SILVA

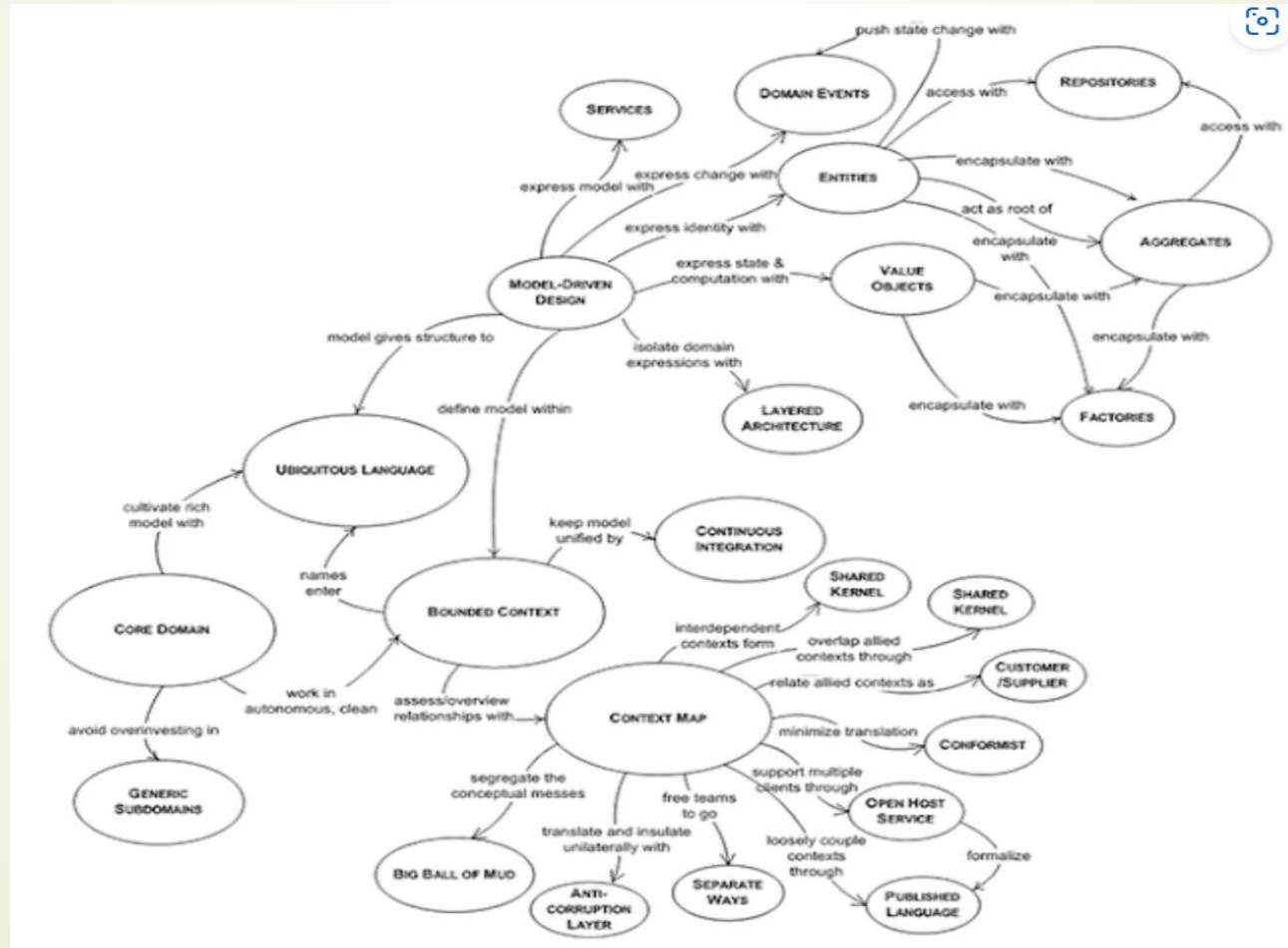
ALUNO: MARLON BRAGA

ALUNO: PEDRO NOVAES

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Mapa de
Aprendizado





Projeto de Bloco: Engenharia de Software e Modelagem

MODELAGEM ESTRATÉGICA

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Pontos Chave

Ubiquitous
Language

Bounded
Context

Context
Map

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Entender a fundo o negócio

Qual problema vai resolver;
O que vai implementar;
Para que serve;
Quem vai atender



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

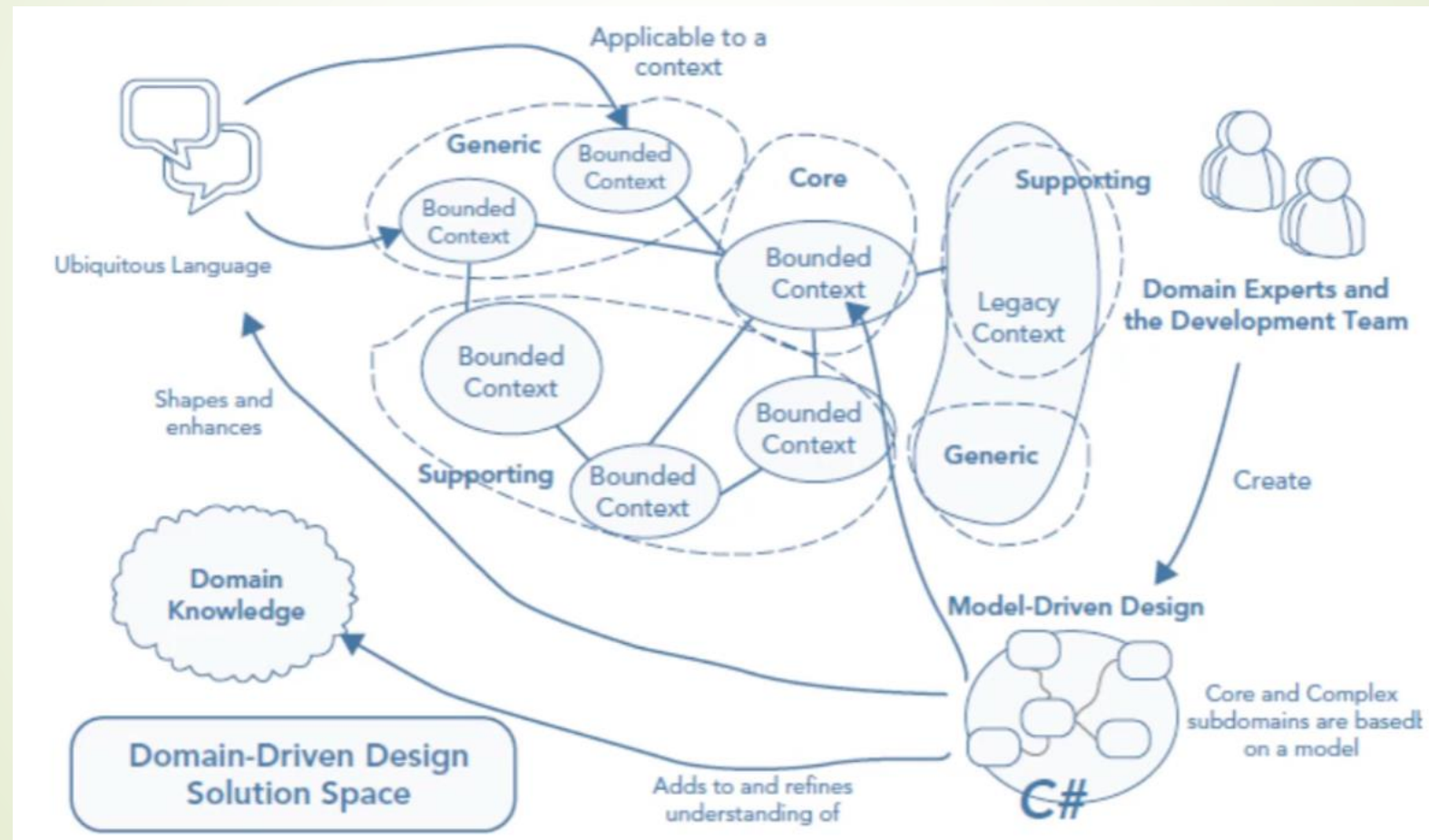
Primeiro passo extrair a linguagem ubíqua

Atenção!!!
Deixar de extrair a
linguagem ubíqua é
um dos piores erros.



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD



- A loja virtual exibirá um catálogo de produtos de diversas categorias.
- Um cliente pode realizar um pedido contendo 1 ou N produtos.
- A loja realizará as vendas através de pagamento por cartão de crédito.
- O cliente irá realizar o seu cadastro para poder fazer pedidos.
- O cliente irá confirmar o pedido, endereço de entrega, escolher o tipo de frete e realizar o pagamento.
- Após o pagamento o pedido mudará de status conforme resposta da transação via cartão.
- Ocorrerá a emissão da nota fiscal logo após a confirmação de pagamento do pedido.

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD



- A loja virtual exibirá um **catálogo** de **produtos** de diversas **categorias**.
- Um **cliente** pode realizar um **pedido** contendo 1 ou N produtos.
- A loja realizará as **vendas** através de **pagamento** por **cartão de crédito**.
- O cliente irá realizar o seu **cadastro** para poder fazer pedidos.
- O cliente irá confirmar o pedido, **endereço** de entrega, escolher o tipo de **frete** e realizar o pagamento.
- Após o pagamento o pedido mudará de **status** conforme resposta da **transação** via cartão.
- Ocorrerá a emissão da **nota fiscal** logo após a confirmação de pagamento do pedido.

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD



- A loja virtual **exibirá um catálogo de produtos de diversas categorias.**
- Um cliente pode **realizar um pedido** contendo 1 ou N produtos.
- A loja **realizará as vendas** através de **pagamento por cartão de crédito.**
- O cliente irá **realizar o seu cadastro** para poder fazer pedidos.
- O cliente irá **confirmar o pedido**, endereço de entrega, **escolher o tipo de frete** e **realizar o pagamento.**
- Após o pagamento **o pedido mudará de status** conforme **resposta da transação via cartão.**
- Ocorrerá a **emissão da nota fiscal** logo após a **confirmação de pagamento** do pedido.

Projeto de Bloco: Engenharia de Software e Modelagem

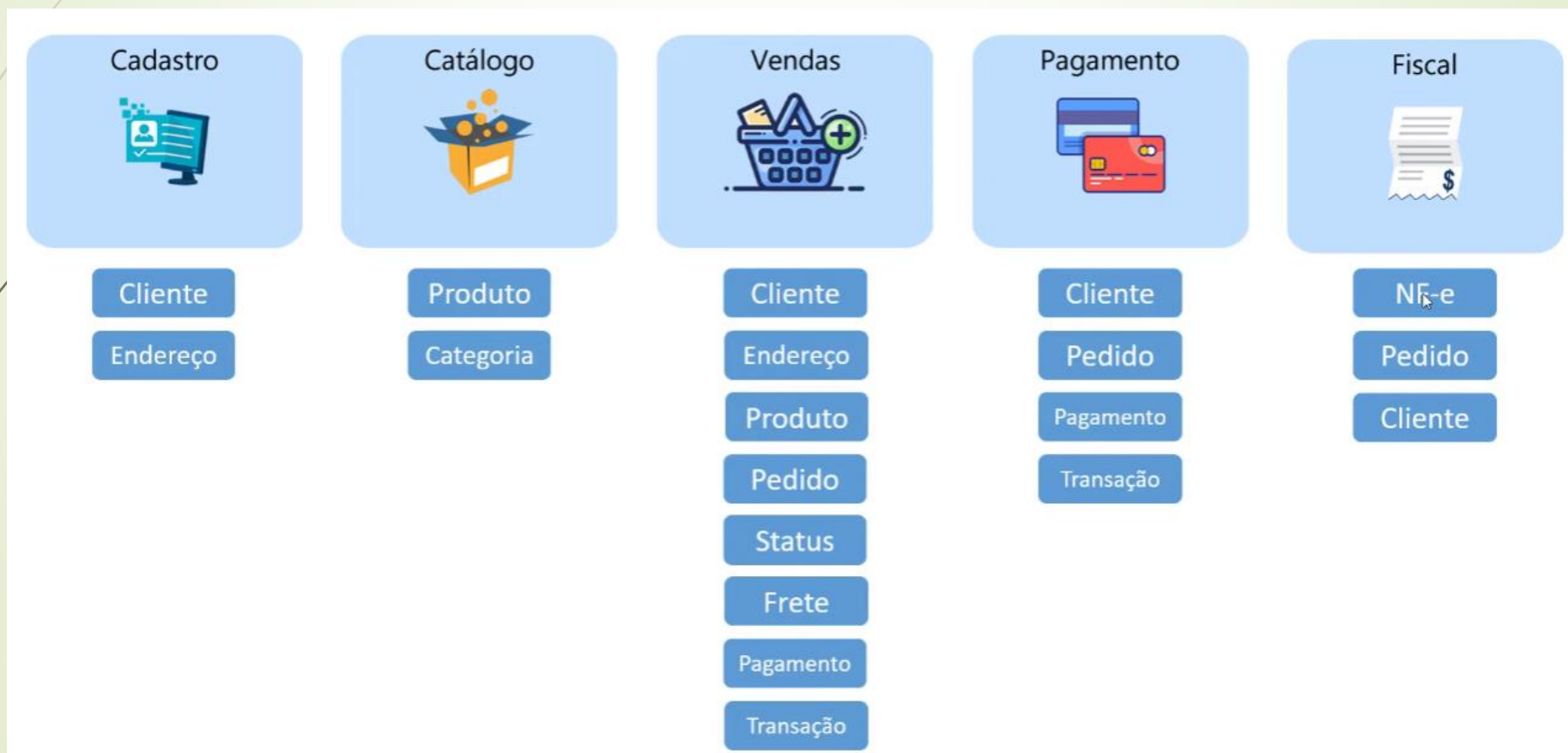
Aplicar os conceitos de DDD

Tipos de Domínios



Projeto de Bloco: Engenharia de Software e Modelagem

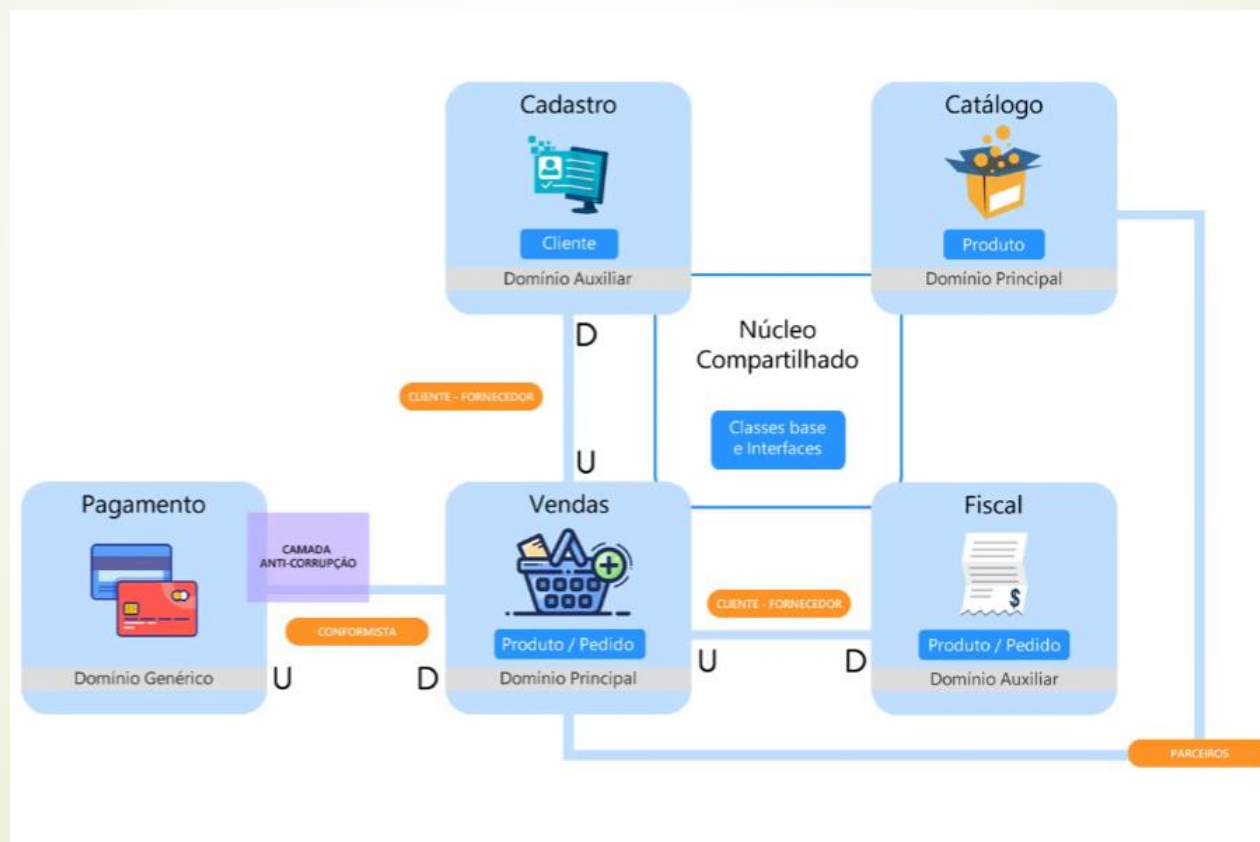
Aplicar os conceitos de DDD



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

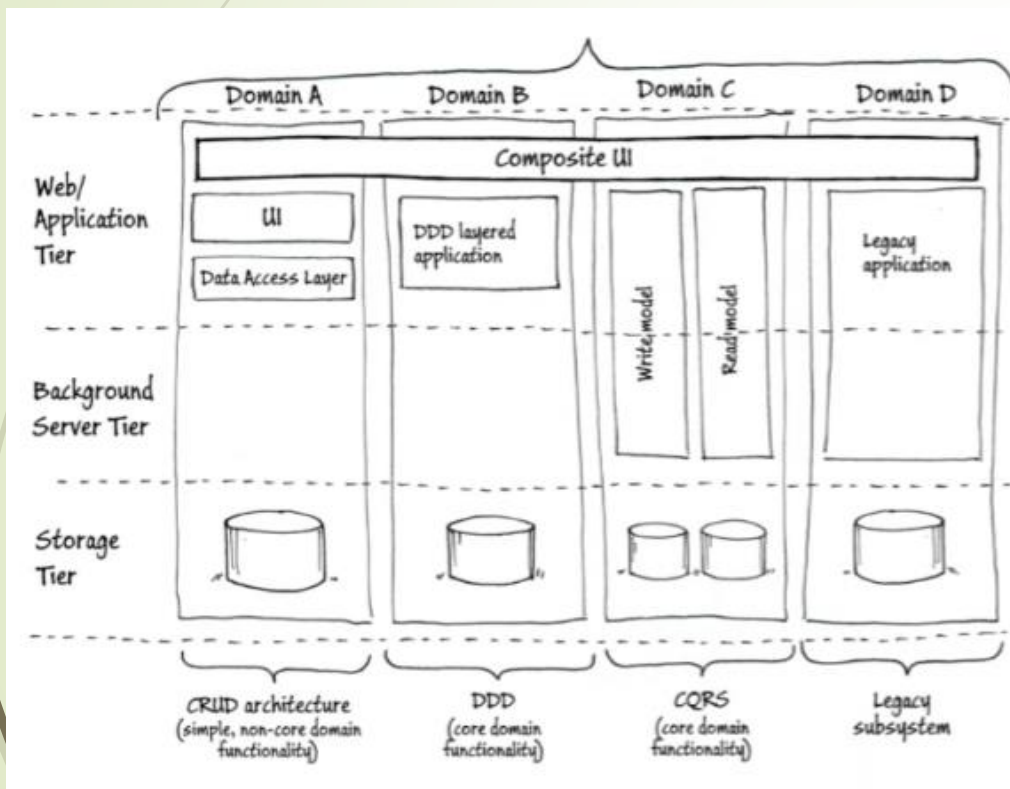
Mapa de Contextos



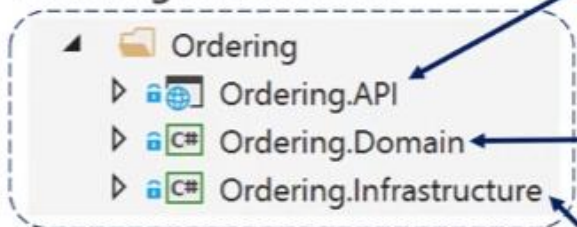
Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Definir arquitetura dos contextos



Ordering microservice



Application layer

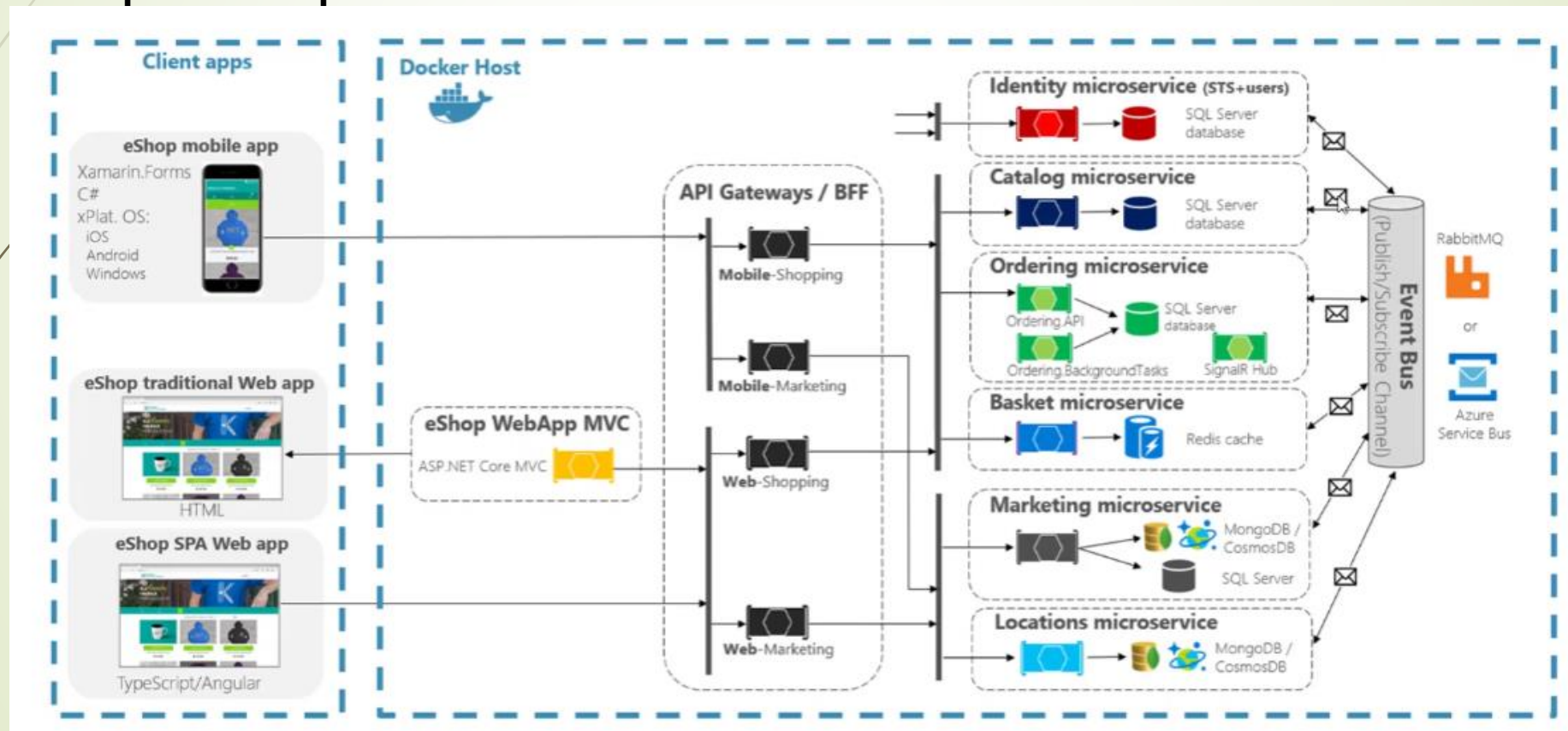
Domain model layer

Infrastructure layer

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Exemplo de arquitetura dos contextos





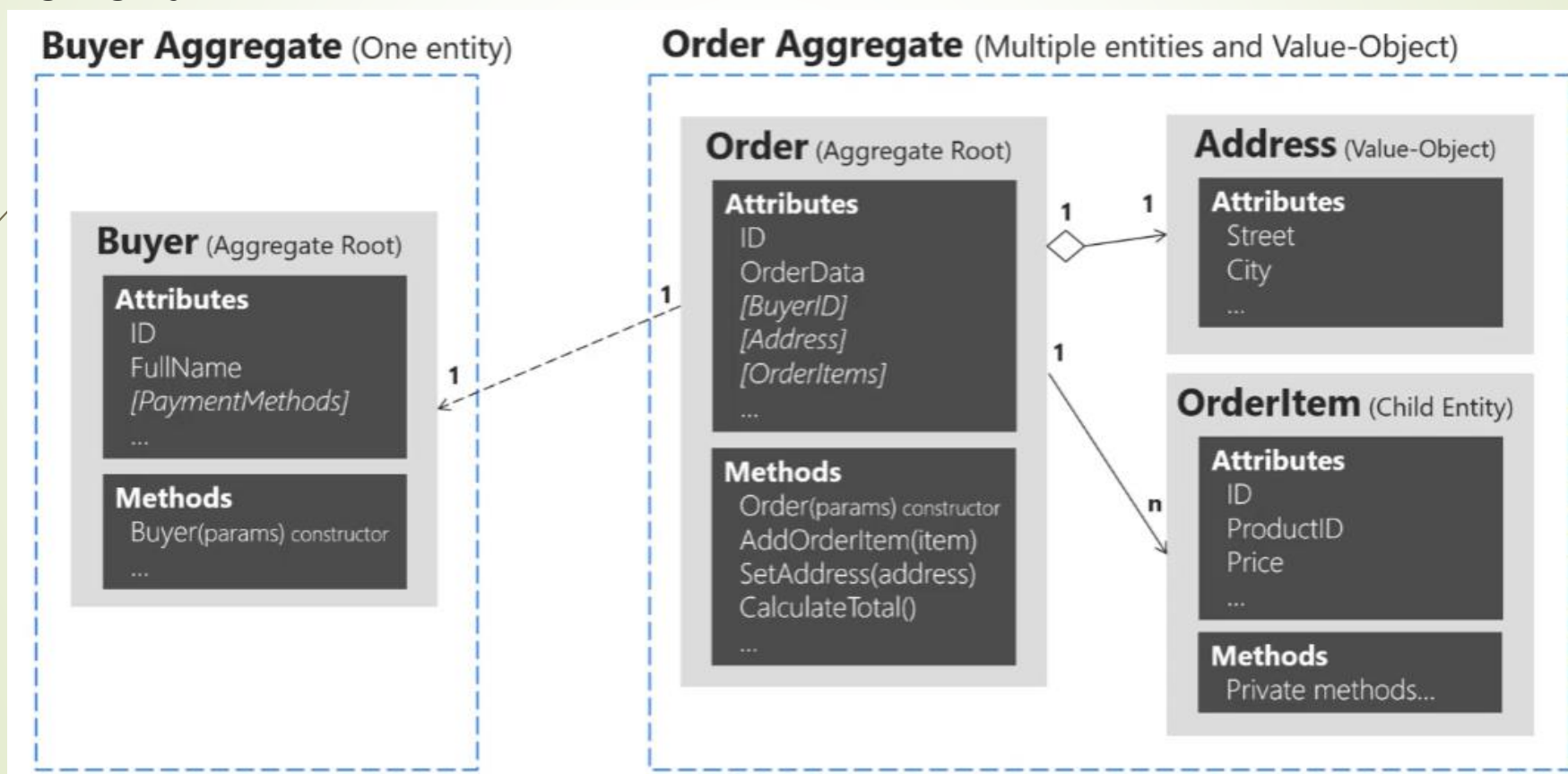
Projeto de Bloco: Engenharia de Software e Modelagem

MODELAGEM TÁTICA

Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

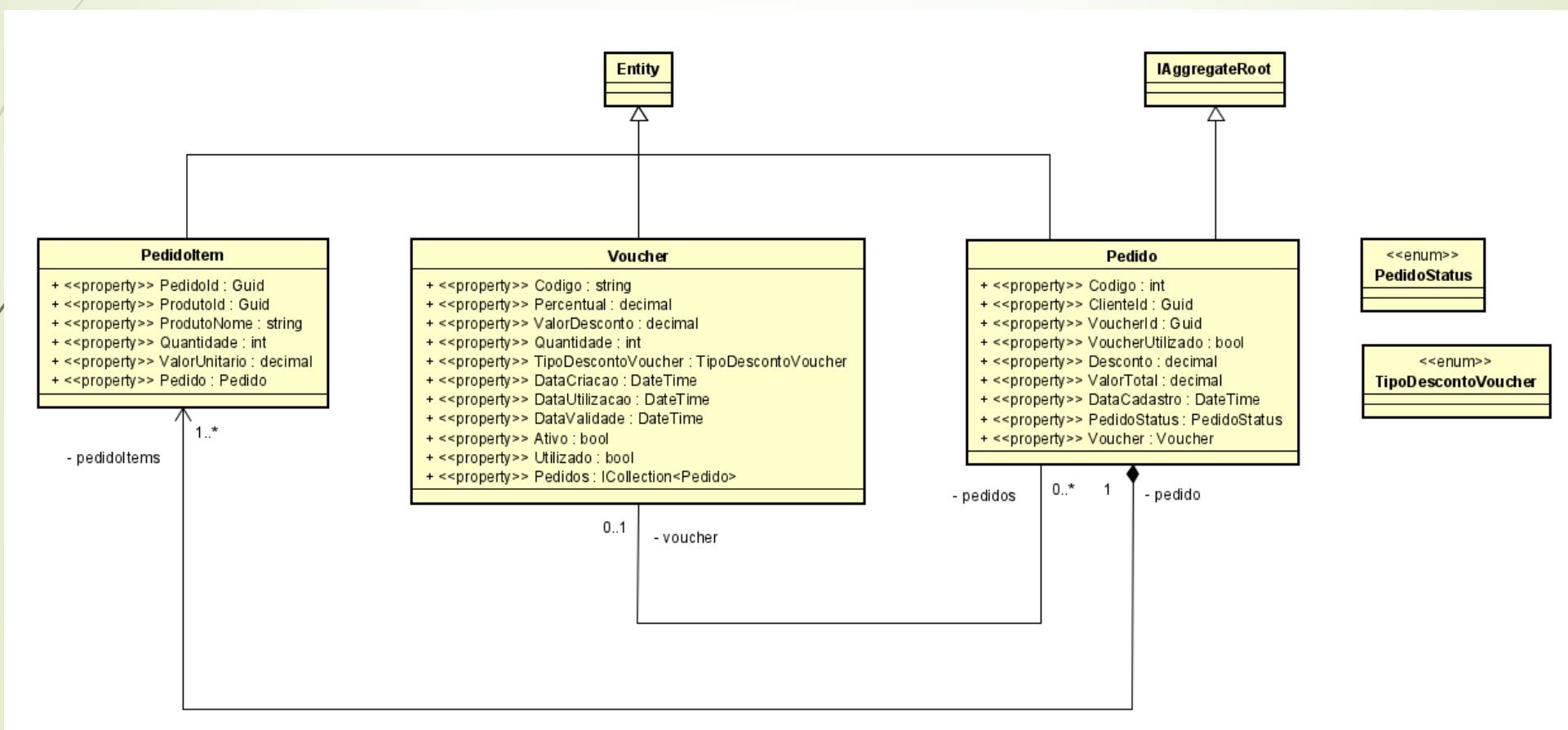
Agregação



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

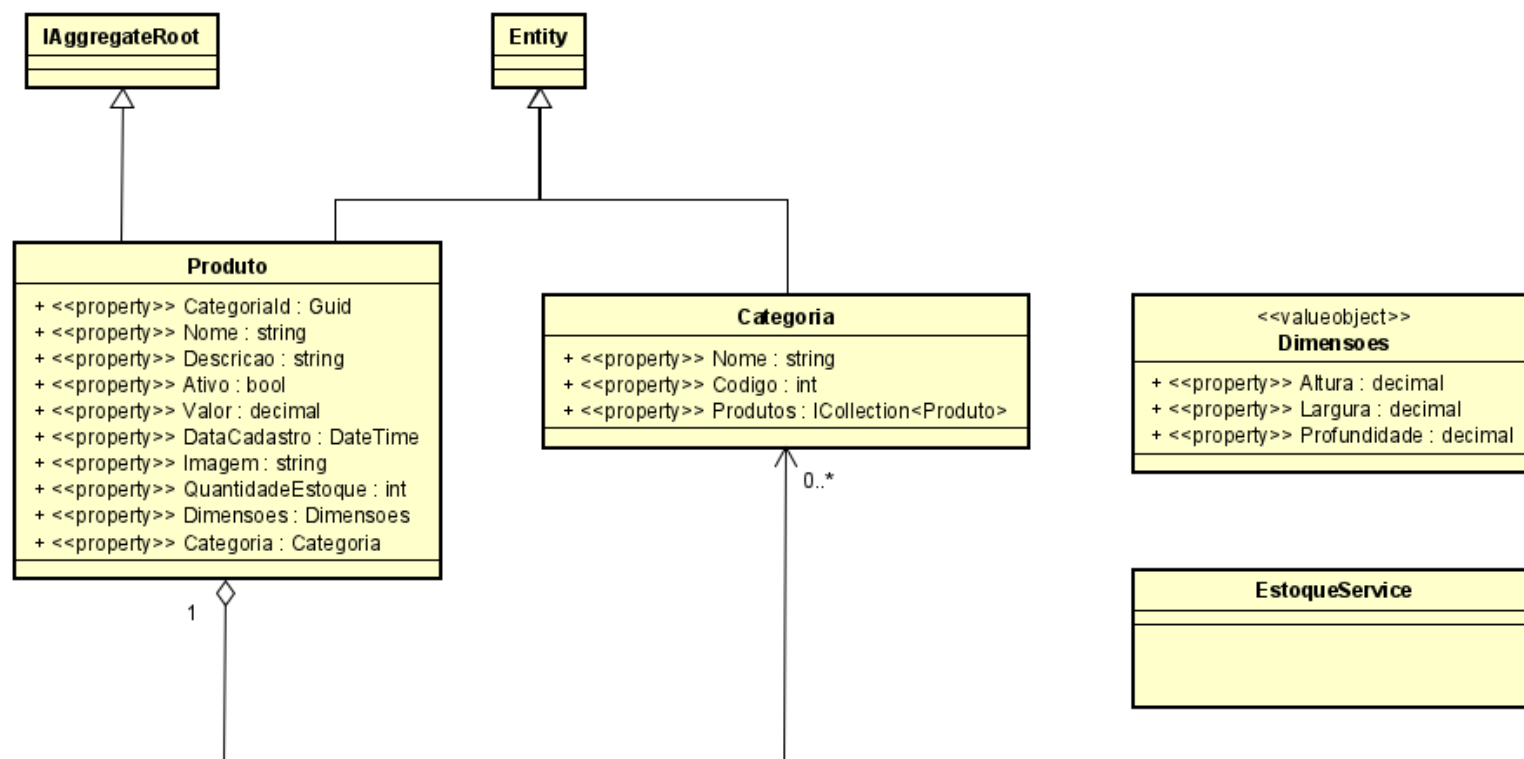
Modelo Conceitual – Contexto Vendas



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Modelo Conceitual – Contexto Catálogo



Projeto de Bloco: Engenharia de Software e Modelagem

Aplicar os conceitos de DDD

Interface de marcação

```
1 namespace NerdStore.Core.DomainObjects
2 {
3     4 referências
4     public interface IAggregateRoot { }
```

```
4 namespace NerdStore.Core.Data
5 {
6     3 referências
7     public interface IRepository<T> : IDisposable where T : IAggregateRoot
8     {
9         18 referências
10        IUnitOfWork UnitOfWork { get; }
```

Raiz de agregação

```
7 namespace NerdStore.Vendas.Domain
8 {
9     21 referências
10    public class Pedido : Entity, IAggregateRoot
11    {
12        3 referências
13        public int Codigo { get; private set; }
14        8 referências
15        public Guid ClienteId { get; private set; }
16        3 referências
17        public Guid? VoucherId { get; private set; }
18        3 referências
19        public bool VoucherUtilizado { get; private set; }
20        4 referências
21        public decimal Desconto { get; private set; }
22        8 referências
23        public decimal ValorTotal { get; private set; }
24        1 referência
25        public DateTime DataCadastro { get; private set; }
26        8 referências
27        public PedidoStatus PedidoStatus { get; private set; }
28
29        private readonly List<PedidoItem> _pedidoItems;
30        10 referências
31        public IReadOnlyCollection<PedidoItem> PedidoItems => _pedidoItems;
32
33        // EF Rel.
34        9 referências
35        public Voucher Voucher { get; private set; }
```

Projeto de Bloco: Engenharia de Software e Modelagem

➤ Referências

DOMINANDO-OS-TESTES-DE-SOFTWARE. **desenvolvedor.io**. Disponível em: <<https://desenvolvedor.io/curso/dominando-os-testes-de-software>>. Acesso em: 16 abr. 2022.

TDD com xUnit para C#.NET Core. **udemy.com**. Disponível em: <<https://www.udemy.com/course/automatizando-testes-para-sua-aplicacao/learn/lecture/9446240?start=0#overview>>. Acesso em: 07 fev. 2019.

C# de teste de unidade com MSTest e .NET. **microsoft.com**. Disponível em: <<https://learn.microsoft.com/pt-br/dotnet/core/testing/unit-testing-with-mstest>>. Acesso em: 26 dez. 2022.

C# de teste de unidade no .NET Core usando dotnet test e xUnit. **microsoft.com**. Disponível em: <<https://learn.microsoft.com/pt-br/dotnet/core/testing/unit-testing-with-dotnet-test>>. Acesso em: 26 dez. 2022.

TESTE de unidade em C# com NUnit e .NET Core. **microsoft.com**. Disponível em: <<https://learn.microsoft.com/pt-br/dotnet/core/testing/unit-testing-with-nunit>>. Acesso em: 26 dez. 2022.

NUNIT vs. XUNIT vs. MSTEST: Comparing Unit Testing Frameworks In C#. **lambdatest.com**. Disponível em: <<https://www.lambdatest.com/blog/nunit-vs-xunit-vs-mstest/>>. Acesso em: 03 dez. 2022.

TESTES de Unidade - Comparando NUnit, XUnit e Visual Studio. **macoratti.net**. Disponível em: <https://www.macoratti.net/20/12/net_unitestc1.htm>. Acesso em: 03 dez. 2022.