

1. Microserviço de Gerenciamento de Clientes

Entidades Principais:

- **Cliente** (**Cliente**)
 - **id** (UUID) - Identificador único
 - **nome** (String)
 - **email** (String)
 - **telefone** (String)
 - **endereco** (String)
 - **dataCadastro** (DateTime)
 - **Referências:**
 - Cada cliente pode ter **vários pedidos** no sistema de pedidos (**id_cliente** referenciado no microserviço de pedidos).
-

2. Microserviço de Catálogo de Produtos

Entidades Principais:

- **Produto** (**Produto**)
 - **id** (UUID) - Identificador único
 - **nome** (String)
 - **descricao** (String)
 - **preco** (Decimal)
 - **quantidadeEstoque** (Integer)
 - **categoria** (String)
 - **Referências:**
 - Cada pedido pode conter **vários produtos** (mas essa relação será gerenciada pelo microserviço de pedidos).
-

3. Microserviço de Gestão de Pedidos

Entidades Principais:

- **Pedido** (**Pedido**)
 - **id** (UUID) - Identificador único
 - **id_cliente** (UUID) - Referência ao Cliente (via API, não FK no BD)

- **status** (Enum) - (CRIADO, PROCESSANDO, ENVIADO, ENTREGUE, CANCELADO)
 - **dataCriacao** (DateTime)
 - **total** (Decimal)
 - **PedidoProduto** (Entidade de relacionamento entre pedidos e produtos)
 - **id** (UUID)
 - **id_pedido** (UUID) - Referência ao Pedido
 - **id_produto** (UUID) - Referência ao Produto (via API, não FK no BD)
 - **quantidade** (Integer)
 - **precoUnitario** (Decimal)
 - **Pagamento** (**Pagamento**) (*Opcional, caso o sistema inclua essa lógica*)
 - **id** (UUID)
 - **id_pedido** (UUID) - Referência ao Pedido
 - **metodoPagamento** (Enum) - (CARTÃO, BOLETO, PIX)
 - **status** (Enum) - (PENDENTE, APROVADO, REJEITADO)
 - **dataPagamento** (DateTime)
 - **Referências:**
 - O pedido precisa do **id_cliente** do serviço de clientes.
 - O pedido precisa do **id_produto** do serviço de catálogo.
 - O status do pedido pode impactar na logística de entrega.
-

4. Microsserviço de Logística de Entrega

Entidades Principais:

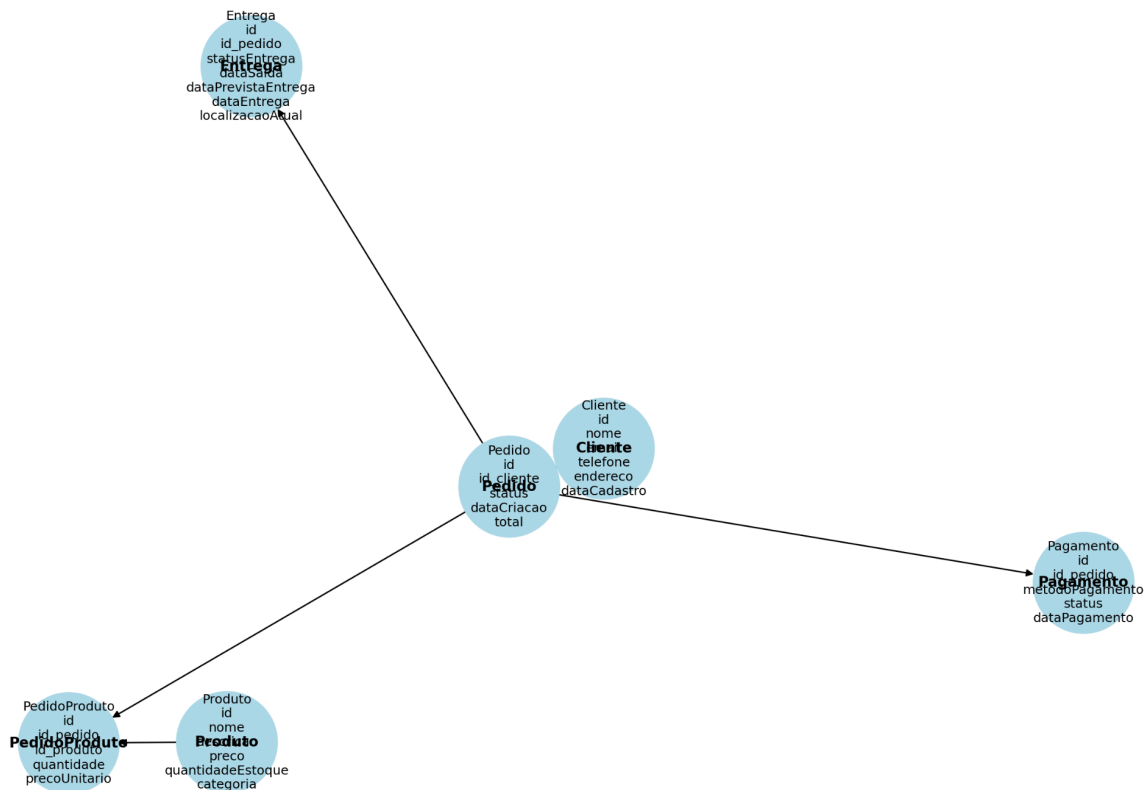
- **Entrega** (**Entrega**)
 - **id** (UUID) - Identificador único
 - **id_pedido** (UUID) - Referência ao Pedido
 - **statusEntrega** (Enum) - (AGUARDANDO, EM ROTA, ENTREGUE, CANCELADO)
 - **dataSaida** (DateTime)
 - **dataPrevistaEntrega** (DateTime)
 - **dataEntrega** (DateTime, nullable)
 - **localizacaoAtual** (String, nullable)
- **Referências:**
 - Cada entrega está associada a um **pedido** (**id_pedido** do microsserviço de pedidos).
 - O status da entrega pode impactar a **atualização do status do pedido**.

Integração Entre Microserviços

Como cada microserviço tem seu próprio banco, a referência entre entidades será feita via APIs REST ou **eventos assíncronos** com **Spring Cloud Stream**:

- ✓ **Clientes** → **Pedidos**: Quando um cliente faz um pedido, seu **id** é armazenado no microserviço de pedidos.
- ✓ **Pedidos** → **Produtos**: O microserviço de pedidos consulta os produtos e armazena os **id_produto**.
- ✓ **Pedidos** → **Logística**: Quando um pedido é aprovado, um evento é disparado para a logística.
- ✓ **Logística** → **Pedidos**: Quando uma entrega é concluída, um evento é enviado para atualizar o pedido.

Diagrama Entidade-Relacionamento (ER)



```
import jakarta.persistence.*;
import java.util.*;
```

@Entity

```
public class Cliente {
    @Id @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;
    private String nome;
    private String email;
    private String telefone;
    private String endereco;
    private Date dataCadastro;
}
```

@Entity

```
public class Produto {
    @Id @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;
    private String nome;
    private String descricao;
    private double preco;
    private int quantidadeEstoque;
    private String categoria;
}
```

@Entity

```
public class Pedido {
    @Id @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;
    private UUID idCliente; // Referência ao Cliente
    private String status;
    private Date dataCriacao;
    private double total;
}
```

@Entity

```
public class PedidoProduto {
    @Id @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;
    private UUID idPedido;
    private UUID idProduto;
    private int quantidade;
    private double precoUnitario;
}
```

```
@Entity
public class Pagamento {
    @Id @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;
    private UUID idPedido;
    private String metodoPagamento;
    private String status;
    private Date dataPagamento;
}
```

```
@Entity
public class Entrega {
    @Id @GeneratedValue(strategy = GenerationType.UUID)
    private UUID id;
    private UUID idPedido;
    private String statusEntrega;
    private Date dataSaida;
    private Date dataPrevistaEntrega;
    private Date dataEntrega;
    private String localizacaoAtual;
}
```

// JSON Requests

```
// Cliente
// POST /clientes
{
    "nome": "João Silva",
    "email": "joao@email.com",
    "telefone": "99999-9999",
    "endereco": "Rua das Flores, 123",
    "dataCadastro": "2024-01-18T10:00:00Z"
}
```

```
// Produto
// POST /produtos
{
    "nome": "Laptop X",
    "descricao": "Laptop potente com 16GB RAM",
    "preco": 4500.00,
    "quantidadeEstoque": 50,
    "categoria": "Eletrônicos"
}
```

```
// Pedido
// POST /pedidos
{
  "idCliente": "UUID_DO_CLIENTE",
  "status": "CRIADO",
  "dataCriacao": "2024-01-18T10:05:00Z",
  "total": 4500.00
}

// PedidoProduto
// POST /pedidos/{idPedido}/produtos
{
  "idProduto": "UUID_DO_PRODUTO",
  "quantidade": 1,
  "precoUnitario": 4500.00
}

// Pagamento
// POST /pagamentos
{
  "idPedido": "UUID_DO_PEDIDO",
  "metodoPagamento": "CARTÃO",
  "status": "PENDENTE",
  "dataPagamento": "2024-01-18T10:10:00Z"
}

// Entrega
// POST /entregas
{
  "idPedido": "UUID_DO_PEDIDO",
  "statusEntrega": "AGUARDANDO",
  "dataSaida": "2024-01-18T12:00:00Z",
  "dataPrevistaEntrega": "2024-01-20T18:00:00Z",
  "localizacaoAtual": "Centro de Distribuição"
}
```