

Projeto de banco de dados

SwiftieStore - Loja online de camisetas

Equipe

Arthur Alves

@alfa

Gabrielle Almeida

@gao2

Júlio Bem

@jcsb2

Penélope Araújo

@pmpa

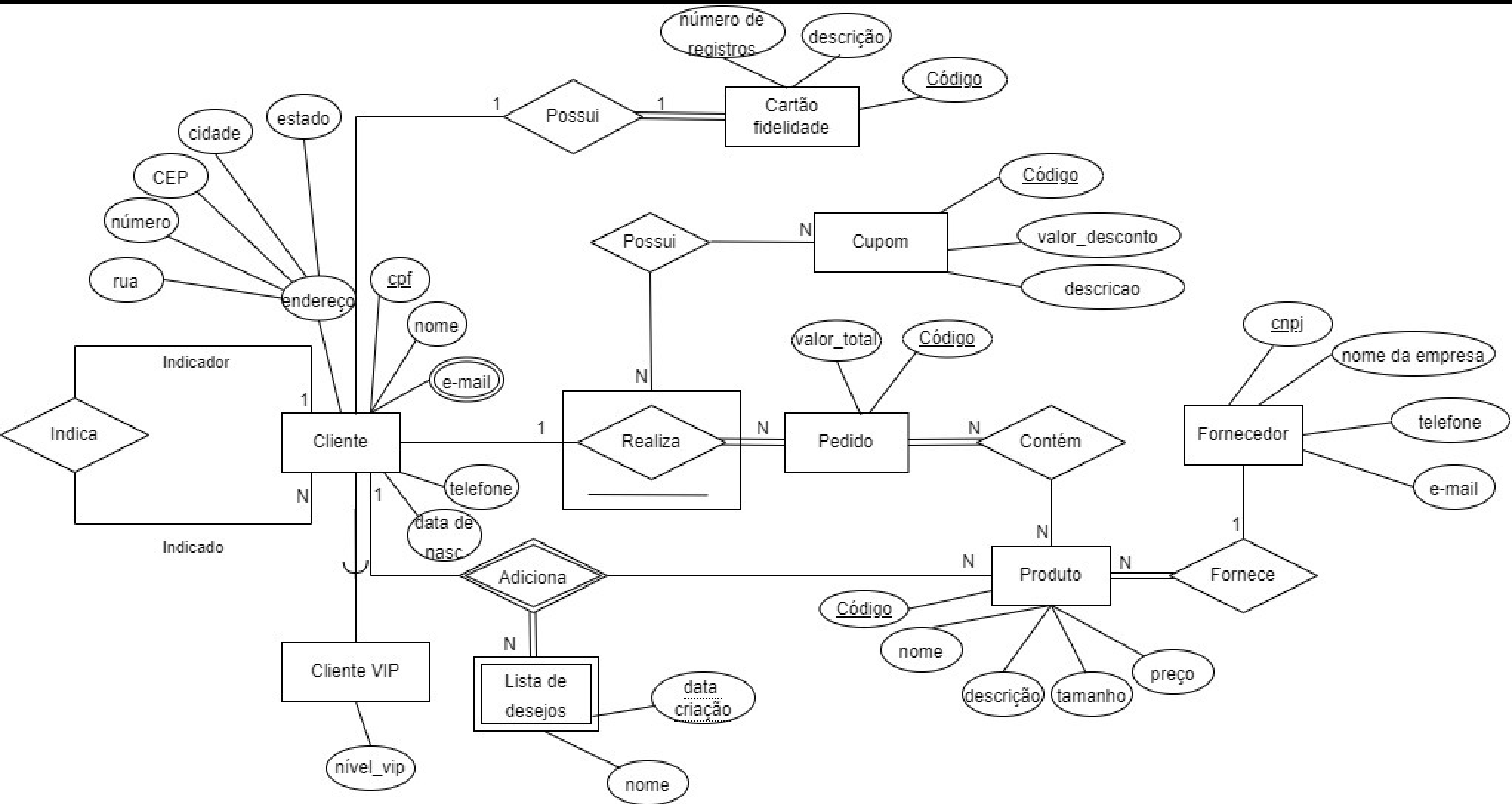
01 - Contexto

Loja de Camisetas

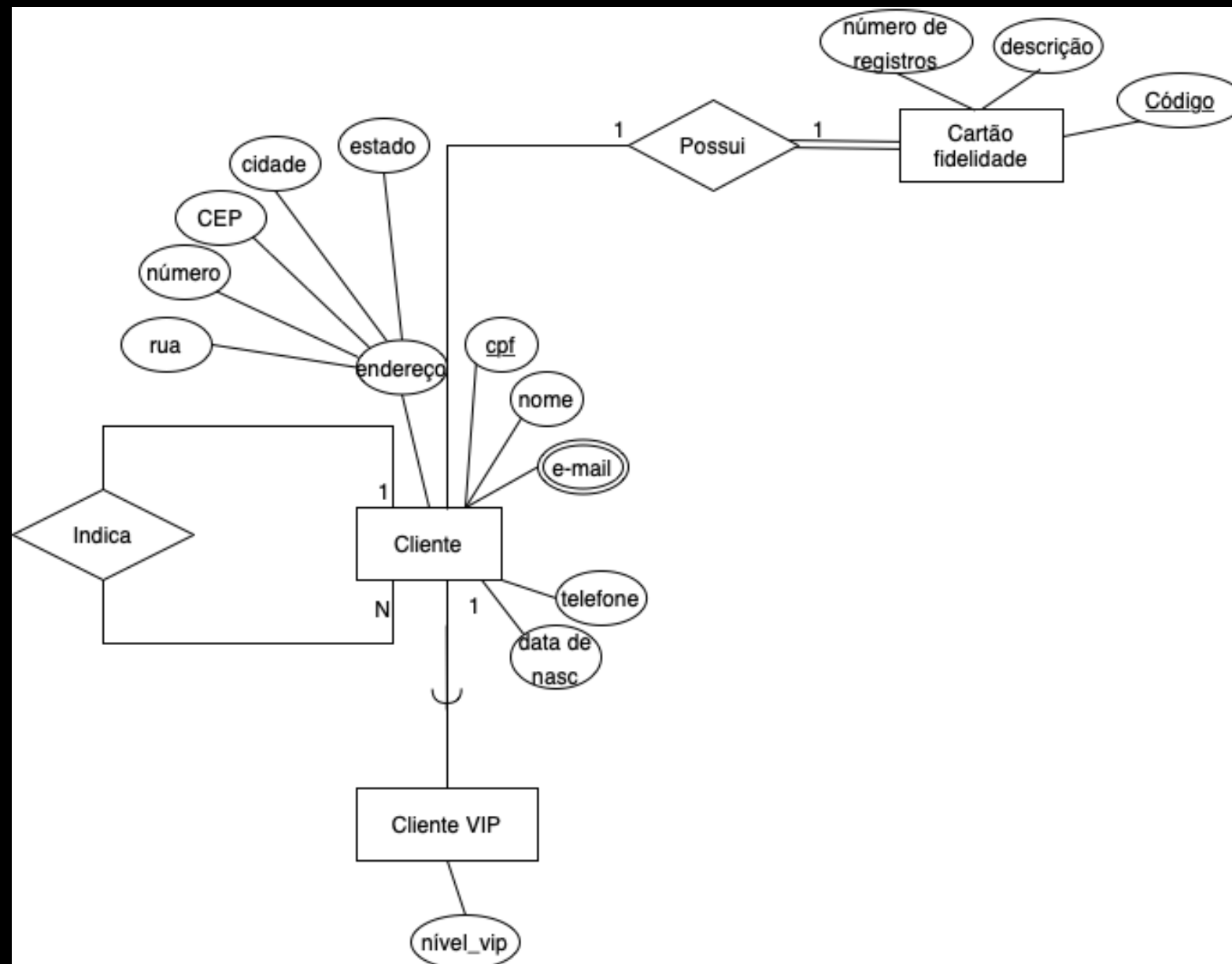


Uma loja virtual dedicada a produção de camisetas. Clientes fiéis, produtos exclusivos, fornecedores de confiança e promoções especiais, tudo isso em um único destino para os fãs da cantora Taylor Swift.

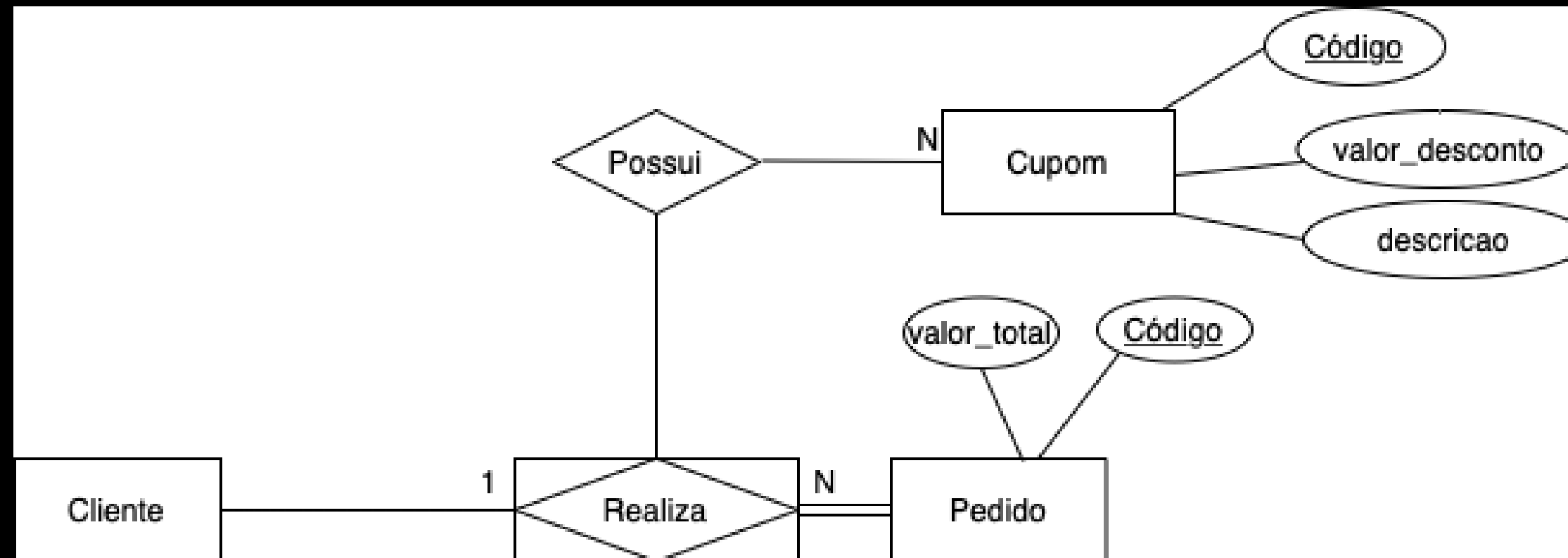
02 - Projeto Conceitual



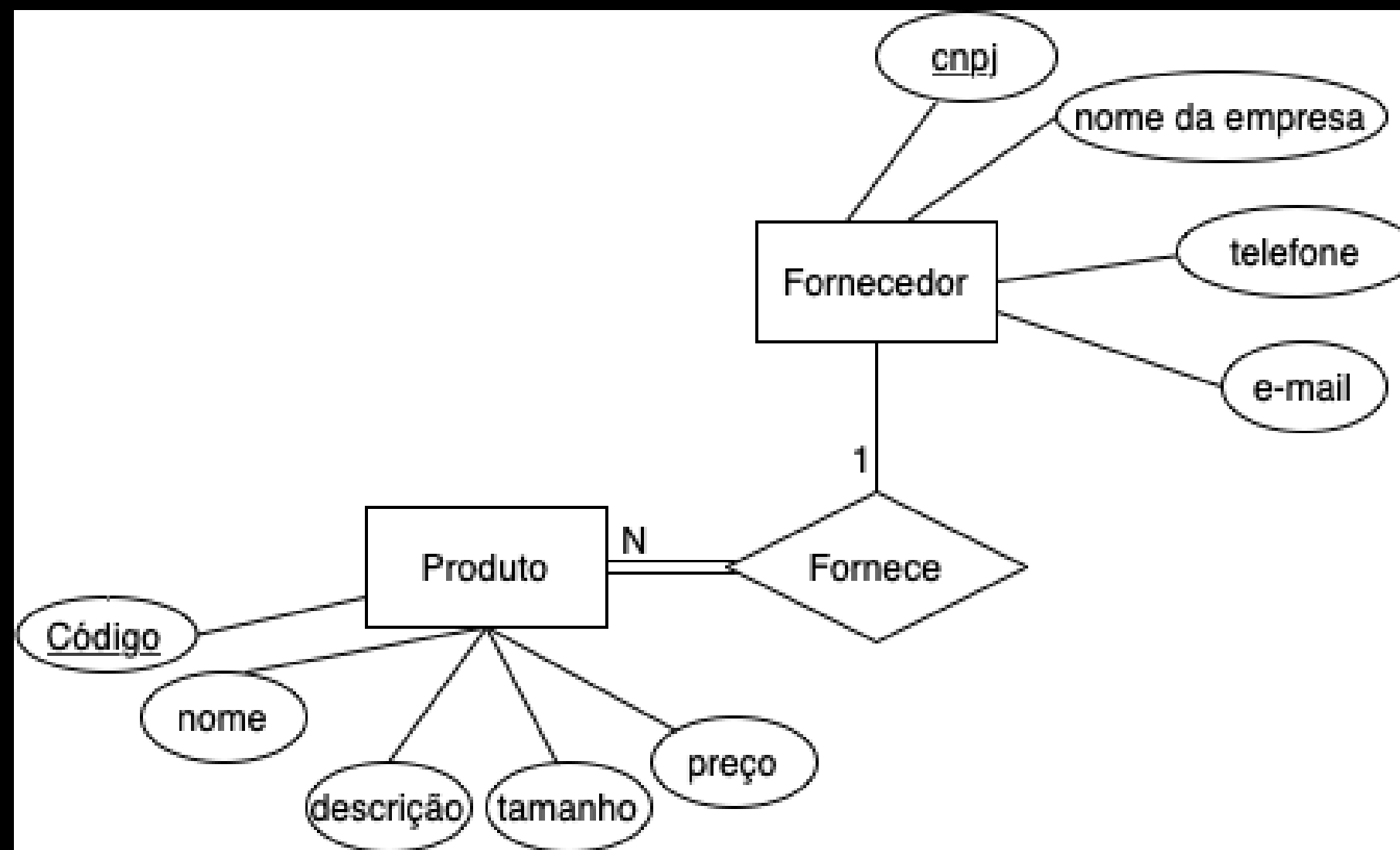
03 - Projeto Lógico



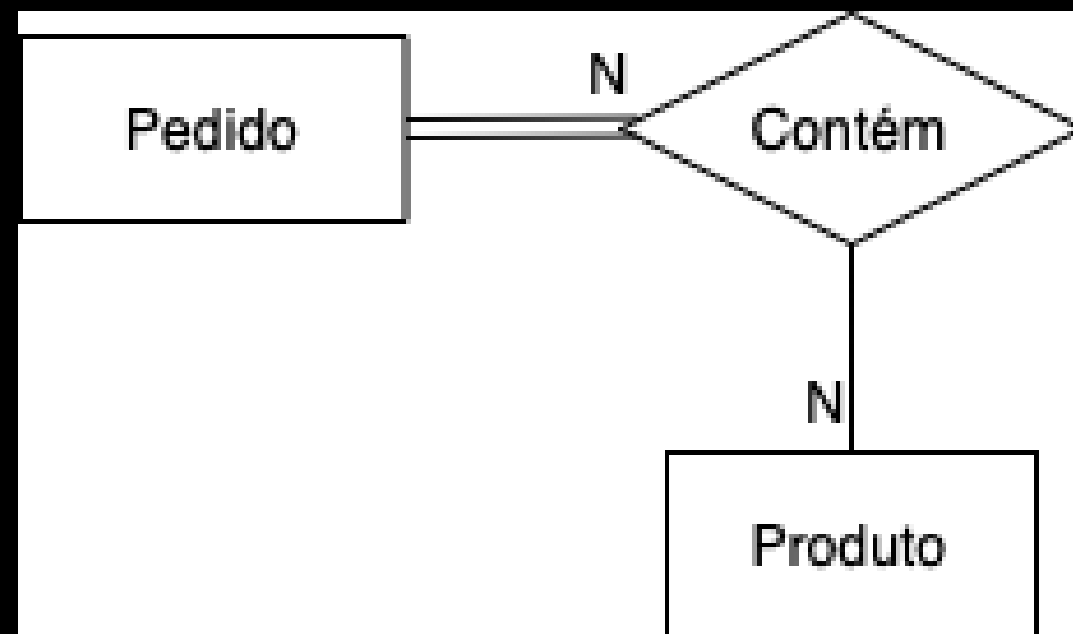
- Cliente(cpf, nome, telefone, data_nasc, end_rua, end_numero, end_cep, end_cidade, end_estado, eh_vip, nivel_vip, cpf_indicador)
 - cpf_indicador -> Cliente(cpf)
- Email(cpf, email)
 - cpf -> Cliente(cpf)
- CartaoFidelidade(cod_cartao, [cod_cliente]!, descricao, num_registros)
 - cod_cliente -> Cliente(cpf)



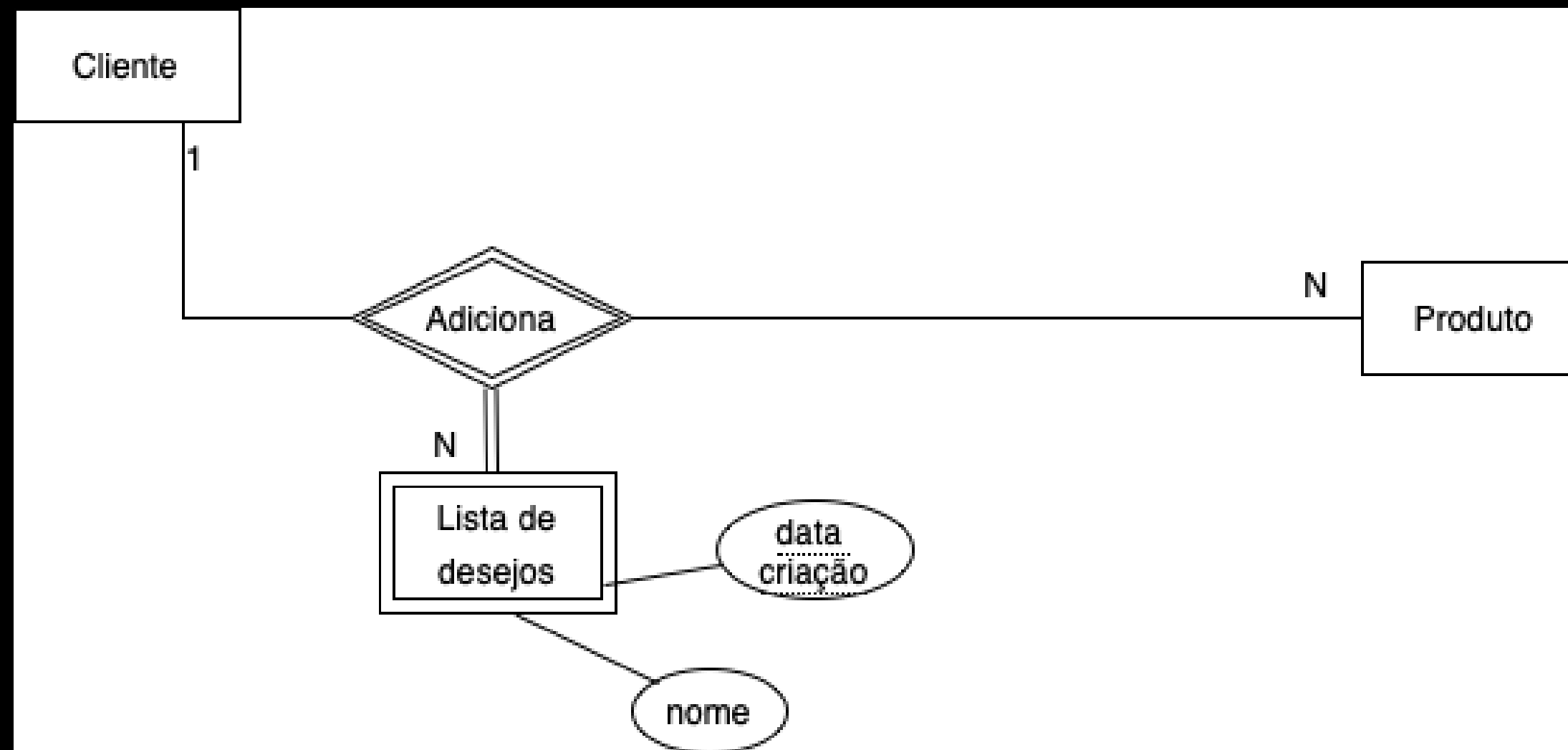
- Pedido(cod_pedido, cod_cliente!, valor_total)
 - cod_cliente -> Cliente(cpf)
- Cupom(cod_promo, valor_desconto, descricao)
- RealizaCompra(cpf, cod_pedido)
 - cpf -> Cliente(cpf)
 - cod_pedido -> Pedido(cod_pedido)
- RealizaCompraCupom(cpf, cod_pedido, cod_cupom)
 - cpf -> Cliente(cpf)
 - cod_pedido -> Pedido(cod_pedido)
 - cod_cupom -> Cupom(cod_promo)



- Produto(cod_produto, nome, descricao, tamanho, preço, cod_fornecedor!)
- cod_fornecedor -> Fornecedor(cnpj)
- Fornecedor(cnpj, nome, telefone, email)



- Contém(cod_pedido, cod_produto)
 - cod_pedido -> Pedido(cod_pedido)
 - cod_produto -> Produto(cod_produto)



- ListaDesejos(cpf, data_criacao, nome)
- Adiciona(cpf!, cod_produto, cod_listaDesejos)
 - cpf -> Cliente(cpf)
 - cod_produto -> Produto(cod_produto)
 - cod_listaDesejos -> ListaDesejos(cpf, data_criacao)

04 - Projeto Físico

GROUP BY e HAVING

Quais os clientes que fizeram mais de um pedido?



```
SELECT c.nome AS cliente, COUNT(p.cod_pedido) AS total_pedidos
FROM Cliente c
LEFT JOIN Pedido p ON c.cpf = p.cod_cliente
GROUP BY c.nome, c.cpf
HAVING COUNT(p.cod_pedido) > 1;
```

CLIENTE	TOTAL_PEDIDOS
Cliente C	2

INNER JOIN

Listar todos os produtos e seus fornecedores



```
SELECT p.nome AS produto, f.nome AS fornecedor  
FROM Produto p  
INNER JOIN Fornecedor f ON p.cod_fornecedor = f.cnpj;
```

PRODUTO	FORNECEDOR
Produto X	Fornecedor A
Produto Y	Fornecedor B
Produto Z	Fornecedor C
Produto W	Fornecedor D

LEFT JOIN

Listar todos os clientes e seus pedidos,
incluindo clientes sem pedidos



```
SELECT c.nome AS cliente, p.cod_pedido  
FROM Cliente c  
LEFT JOIN Pedido p ON c.cpf = p.cod_cliente;
```

CLIENTE	COD_PEDIDO
Cliente A	101
Cliente B	102
Cliente C	103
Cliente C	104
Cliente F	-
Cliente E	-
Cliente D	-

SEMI JOIN

Listar todos os clientes que fizeram pelo menos um pedido



```
SELECT c.cpf, c.nome
FROM Cliente c
WHERE EXISTS (
    SELECT *
    FROM Pedido p
    WHERE p.cod_cliente = c.cpf
)
```

CPF	NOME
111111111111	Cliente A
222222222222	Cliente B
333333333333	Cliente C

ANTI JOIN

Quais os clientes que ainda não fizeram nenhum pedido?



```
SELECT c.cpf, c.nome  
FROM Cliente c  
WHERE NOT EXISTS (  
    SELECT *  
    FROM Pedido p  
    WHERE p.cod_cliente = c.cpf  
)
```

CPF	NOME
666666666666	Cliente F
555555555555	Cliente E
444444444444	Cliente D

SUBCONSULTA ESCALAR

Qual o nome do cliente que fez o pedido mais recente?

```
SELECT c.nome
FROM Cliente c
WHERE c.cpf = (
    SELECT p.cod_cliente
    FROM Pedido p
    WHERE p.cod_pedido = (
        SELECT MAX(p2.cod_pedido)
        FROM Pedido p2
    )
);
```

NOME
Cliente C

SUBCONSULTA LINHA

Quais os detalhes do pedido 101?



```
SELECT p.cod_produto, p.nome, p.descricao
FROM Produto p
WHERE p.cod_produto IN (
    SELECT cod_produto
    FROM Contem
    WHERE cod_pedido = 101
);
```

COD_PRODUTO	NOME	DESCRICAO
1	Produto X	Descrição do Produto X
2	Produto Y	Descrição do Produto Y

SUBCONSULTA TABELA

Quais os pedidos feitos por clientes VIP?



```
SELECT cod_pedido, cod_cliente
FROM Pedido
WHERE cod_cliente IN (
    SELECT cpf
    FROM Cliente
    WHERE eh_vip = 1
);
```

COD_PEDIDO	COD_CLIENTE
101	111111111111
103	333333333333
104	333333333333

UNION ALL (op. de conjunto)

Listar todos os clientes e fornecedores cadastrados, com seu respectivo tipo de cadastro



```
SELECT nome AS entidade, 'Cliente' AS tipo FROM CLIENTE
UNION ALL
SELECT nome AS entidade, 'Fornecedor' AS tipo FROM FORNECEDOR;
```

ENTIDADE	TIPO
Cliente A	Cliente
Cliente B	Cliente
Cliente D	Cliente
Cliente E	Cliente
Cliente F	Cliente
Cliente C	Cliente
Fornecedor A	Fornecedor
Fornecedor B	Fornecedor
Fornecedor C	Fornecedor
Fornecedor D	Fornecedor

Consulta alternativa para calcular o valor total de um pedido (substituir 101 pelo código do pedido)



```
SELECT p.cod_pedido,  
       SUM(pr.preco) AS valor_total  
FROM Pedido p  
JOIN Contem c ON p.cod_pedido = c.cod_pedido  
JOIN Produto pr ON c.cod_produto = pr.cod_produto  
WHERE p.cod_pedido = 101  
GROUP BY p.cod_pedido, p.cod_cliente;
```

COD_PEDIDO	VALOR_TOTAL
101	129.98



Obrigado!