

Grupo 26		Tema: 9	
51938	Tatiana Martins	90142	Bernardo Cozac
90147	Diogo Freitas	90247	Diogo Carvalho

Template de exemplo para a realização da Parte 1 do trabalho prático de bases de dados. Este *template* não deverá ser alterado (manter estilo de letra, espaçamentos e margens).

O texto deve ser apresentado em português corrente.

O Clube LEI é uma associação desportiva dedicada à prática de ténis e padel, aberta a sócios e visitantes. Para além das suas atividades desportivas, o clube possui duas áreas de funcionamento complementares, a loja de artigos desportivos e o bar/cafetaria, que desempenham um papel central no apoio aos praticantes e visitantes e representam uma importante fonte de receita.

A loja comercializa artigos especializados para a prática de ténis e padel, como raquetes, bolas, calçado, calcões, t-shirts, bonés, grips e sacos. Cada produto é caracterizado por um conjunto de atributos essenciais, nomeadamente: nome do artigo, categoria (ex.: raquete, vestuário, acessório), marca, modelo, preço de venda, código identificador único e quantidade em stock. O processo de venda está disponível para sócios e não sócios, sendo que os sócios beneficiam de um desconto mediante validação do número de sócio. Cada operação de venda deve registar dados fundamentais como a data e hora da compra, a lista de produtos vendidos e respetivas quantidades, o desconto aplicado, o meio de pagamento (ex.: numerário, cartão) e a identificação do funcionário responsável. A gestão de stock da loja inclui o acompanhamento dos níveis de cada produto e a definição de um valor mínimo por artigo. Quando o stock atinge esse valor, é realizada uma encomenda a fornecedores, operação que deve registar dados como produto, quantidade encomendada, fornecedor e funcionário que a emitiu.

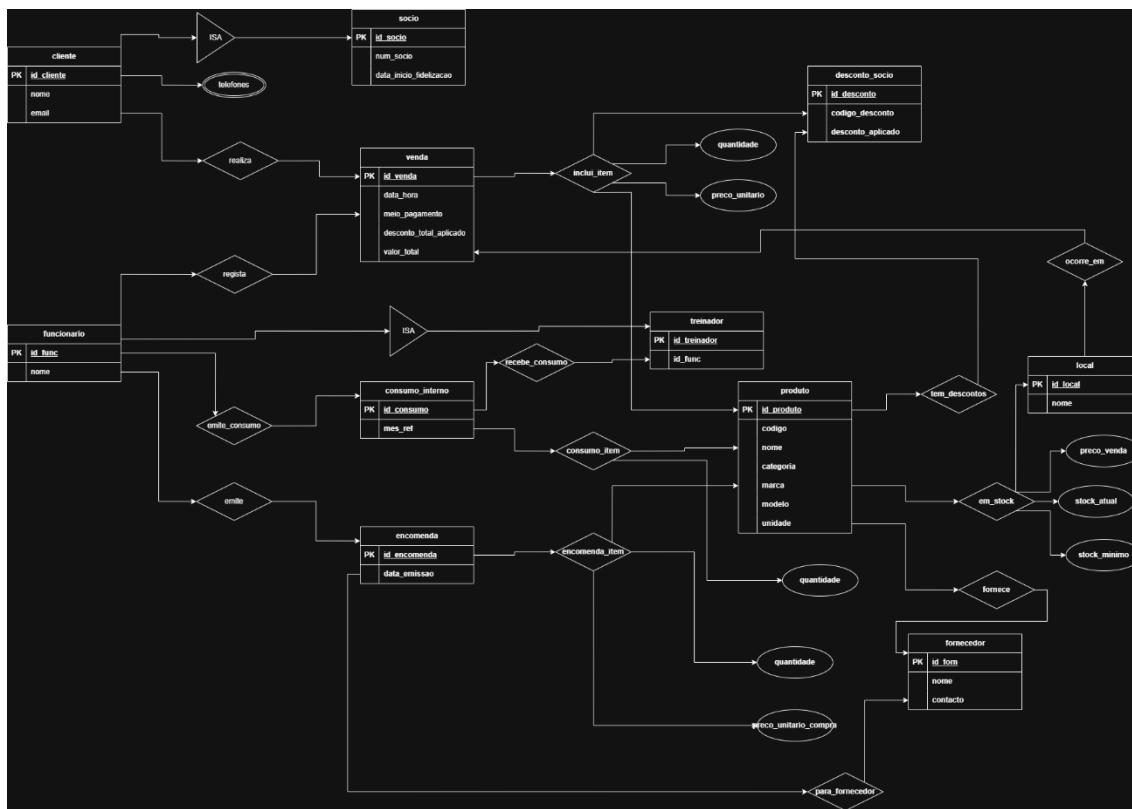
Para além das vendas, a loja também gera o consumo interno de bolas de ténis para as aulas dadas pelos treinadores. Este processo, que decorre no início de cada mês, consiste na entrega de uma quantidade de bolas a cada treinador, de acordo com o plano de treinos. Este movimento deve ser registado com informação como o mês de referência, o nome do treinador e a quantidade de bolas, permitindo que o consumo interno seja abatido ao stock e contabilizado como custo do clube.

O bar/cafetaria disponibiliza produtos de cafetaria (cafés, sumos, bolos, sandes) e refeições rápidas, incluindo pratos do dia, que se distinguem pela sua rotatividade. Tal como na loja, os sócios beneficiam de descontos sobre o preço de venda. Cada operação de venda deve registar dados semelhantes aos da loja: data e hora, produtos consumidos, quantidades, aplicação de desconto e funcionário responsável. A gestão de stock do bar é menos detalhada do que na loja, focando-se sobretudo em produtos finais prontos a servir (bebidas, snacks, refeições) e em matérias-primas essenciais (ex.: café, leite, pão), que devem ser monitorizadas para garantir a reposição atempada.

Face a este enquadramento, torna-se evidente a necessidade de uma base de dados integrada que consolide toda a informação relativa ao funcionamento do clube. Esta base de dados deverá contemplar:

- Gestão de clientes (sócios e não sócios), incluindo identificação, estatuto de sócio [e telefones \(atributo multivalorado\)](#);
- Gestão de produtos (loja e bar), com atributos como nome, categoria, preço, stock e fornecedor; [unidade; código do produto \(EAN/SKU\) como chave candidata; gestão de preço e stock por local \(LOJA/BAR\) via Produto–Local; possibilidade de vários fornecedores por produto \(Produto–Fornecedor\)](#);
- Registo de vendas, com indicação de data, hora, produtos, quantidades, descontos aplicados, meio de pagamento e funcionário responsável; [local da venda; preço unitário capturado na linha; código do desconto aplicado na linha; regra: descontos apenas para sócios \(validação do estatuto\)](#);
- Registo de consumo interno ([de bolas e outros itens](#)), associado a treinador, [mês](#) e [quantidade de bolas](#); [mês de referência \(mes_ref no formato YYYY-MM\) e lista de itens com quantidades](#); [funcionário emitente](#);
- Gestão de stock e respetivos níveis mínimos; [por local \(LOJA/BAR\)](#);
- Gestão de encomendas a fornecedores, com indicação de produto, quantidade e responsável. [Inclui preço unitário de compra na linha](#);
- [Gestão de descontos de sócio por produto \(código de desconto e percentagem/valor associado\)](#);
- [Entidade “Local” para distinguir Loja, Bar e outras secções](#).

Modelo Entidade-Associação (ER)



Restrições de integridade que não são captadas pelo modelo

As restrições de integridade que não ficam explícitas no diagrama são as seguintes: um desconto de sócio só pode ser aplicado quando a venda tem cliente com estatuto de sócio válido na data/hora da venda e o desconto corresponde ao mesmo produto da linha; o preço unitário de cada item de venda é copiado do preço vigente no momento da compra e permanece imutável para preservar o histórico; as quantidades de venda, encomenda e consumo interno são inteiros positivos; os preços e descontos são não negativos e os descontos obedecem ao domínio definido; o stock nunca pode ficar negativo, pelo que movimentos de venda e de consumo interno só se confirmam se, após o débito, o stock continuar ≥ 0 ; cada treinador pode ter no máximo um registo de consumo interno por mês de referência (mes_ref em formato YYYY-MM); os campos temporais não podem apontar para o futuro (a data/hora da venda e a data de emissão da encomenda têm de ser \leq ao momento do registo); e cada item de venda só é válido num local onde exista o par Produto–Local previamente definido (isto é, tem de existir a correspondência entre o produto e o local em catálogo/stock).

Decisões de modelação

As decisões de modelação adotadas foram: representar os locais (Loja, Bar, etc.) como entidade autónoma e controlar stock e preço por local através da associação Produto–Local (em vez de criar subtipos “Produto Loja/Bar”), garantindo escalabilidade e evitando duplicação; modelar os descontos de sócio como entidade própria, aplicados ao nível do item de venda (relação ternária Venda–Produto–Desconto), o que permite auditoria e regras de elegibilidade; armazenar o preço unitário no item de venda para assegurar relatórios estáveis mesmo após alterações de catálogo; representar o consumo interno por mês com o atributo mes_ref (YYYY-MM) e destinatário obrigatório treinadores; considerar telefones como atributo multivalorado do cliente para refletir múltiplos contactos; adotar chaves de negócio como candidatas (por exemplo, produto.codigo — EAN/SKU — e socio.num_socio) mantendo, em paralelo, identificadores técnicos (*id_**) para robustez referencial; especializar Treinador a partir de Funcionário e Sócio a partir de Cliente (especializações parciais e disjuntas), alinhando regras de negócio de consumo e descontos; e representar fornecedor por produto como associação muitos-para-muitos, compatível com múltiplas origens de aprovisionamento.

Modelo relacional

```

cliente(id_cliente, nome, email)
cliente_telefone(id_cliente, telefone)
socio(id_cliente, num_socio, data_inicio_fidelizacao)
funcionario(id_func, nome)
treinador(id_treinador, id_func)
fornecedor(id_forn, nome, contacto)
local(id_local, nome)
produto(id_produto, codigo, nome, categoria, marca, modelo, unidade)
produto_local(id_local, id_produto, stock_atual, stock_minimo, preco_venda)
produto_fornecedor(id_produto, id_forn)
desconto_socio(id_desconto, codigo_desconto, desconto_aplicado, id_produto)
venda(id_venda, data_hora, meio_pagamento, desconto_total_aplicado, id_cliente, id_func, id_local, valor_total)
venda_item(id_venda, id_produto, quantidade, preco_unitario, id_desconto)
encomenda(id_encomenda, data_emissao, id_func, id_forn)
encomenda_item(id_encomenda, id_produto, quantidade, preco_unitario_compra)
consumo_interno(id_consumo, mes_ref, id_func_emitente, id_treinador)
consumo_item(id_consumo, id_produto, quantidade)
  
```

Restrições de integridade

No modelo relacional, a integridade é garantida por chaves primárias em todas as tabelas (compostas nas associativas) e por unicidades de negócio: socio.num_socio, produto.codigo e local.nome são únicos, tal como treinador.id_func para manter o 1:1 com funcionario. A integridade referencial impõe que socio dependa de cliente e treinador de funcionario; os itens eliminam em cascata com as respetivas cabeças (venda_item→venda, encomenda_item→encomenda, consumo_item→consumo_interno); venda referencia obrigatoriamente funcionario e local e opcionalmente cliente, e os itens referenciam produto (obrigatório) e desconto_socio (opcional). Os domínios são validados com CHECK: quantidades de venda/encomenda/consumo são >0 ; preços e descontos são ≥ 0 (se percentagem, em [0,100]); produto_local.stock_atual e stock_minimo são ≥ 0 ; consumo_interno.mes_ref obedece a YYYY-MM e é único por treinador; datas de venda e de emissão de encomenda não podem estar no futuro. As regras de negócio exigem triggers: um desconto só pode ser aplicado se a venda tiver cliente com estatuto de sócio válido à data e se o desconto corresponder ao mesmo produto do item; o preco_unitario do item é copiado do preço vigente em produto_local no momento/local da venda e torna-se imutável; só é permitido inserir um item se existir previamente o par (id_local, id_produto) em produto_local; e movimentos de venda/consumo só confirmam se, após o débito, o stock no local permanecer ≥ 0 .