

Faculdade de Engenharia da Universidade do Porto



Projeto Base de Dados 23/24
Submissão 1

Loja De Roupas Online
Grupo 604

Docente: António Humberto Sá Pinto

Estudantes & Autores:

Beatriz Sonnemberg up202206098@fe.up.pt

Diogo Sousa up202204135@fe.up.pt

Miguel Duarte up2022006102@fe.up.pt

FIRST SUBMISSION

4.1

A- Domain Definition

Pretende-se armazenar a informação relativa a uma Loja de Roupas Online.

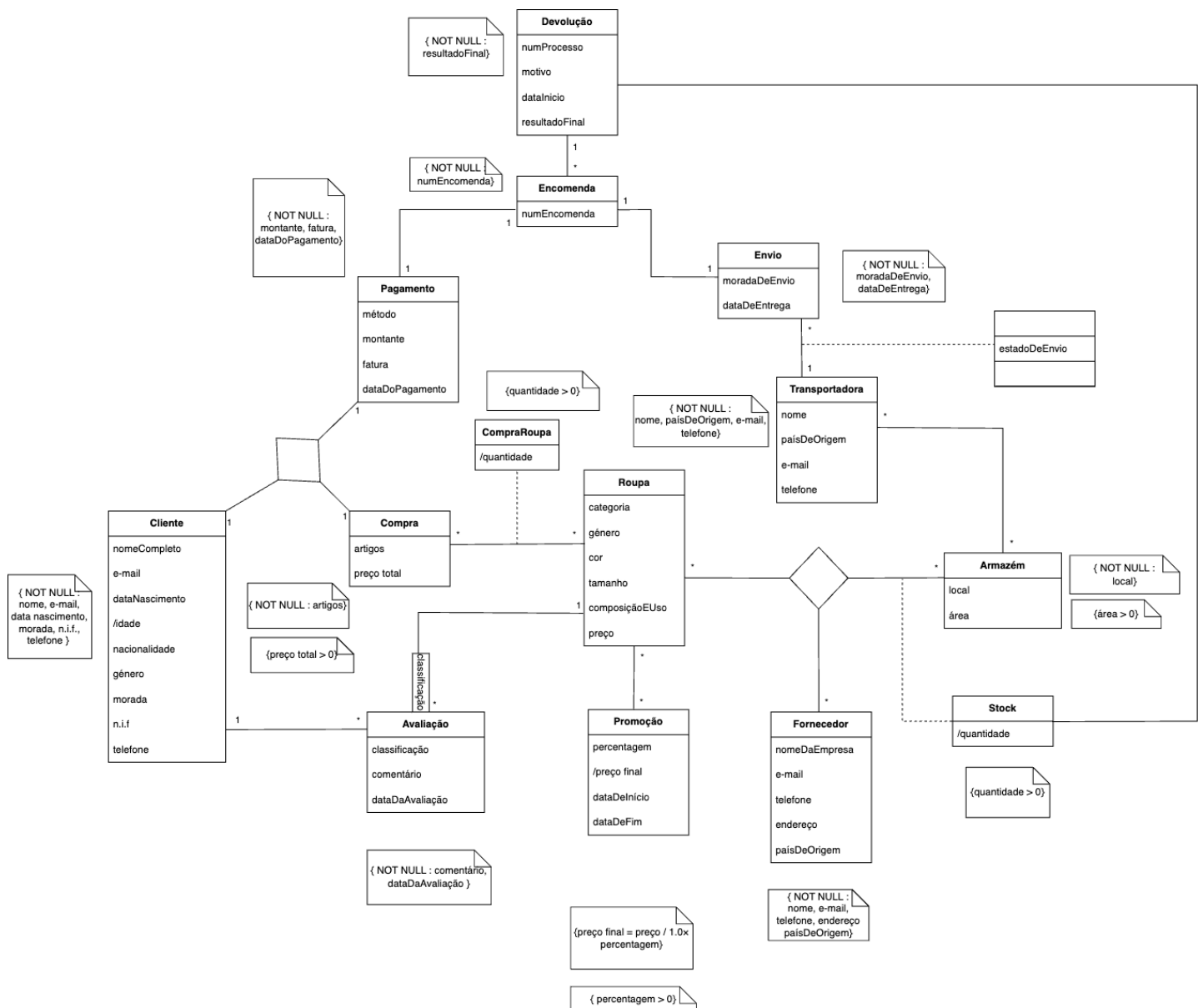
De cada peça de roupa, é necessário saber a sua categoria (casaco, calças, sweatshirt...), a cor, o tamanho, o género e a sua composição e uso. Cada roupa deverá ter uma breve descrição e um preço original. Interessa também saber as avaliações e promoções de cada roupa. Da avaliação, é imperativo saber a sua classificação, o comentário e a data da avaliação. A promoção será definida por uma percentagem que vai gerar um novo preço final. Cada promoção é caracterizada por uma percentagem e um preço final. Além disso, as promoções são também limitadas pelas datas de início e de fim.

Para cada cliente é necessário saber o seu nome completo, e-mail, data de nascimento, nacionalidade, género, telefone, morada e NIF. Os clientes fazem compras que podem ter uma ou várias roupas. Antes de avançar para o pagamento, é imperioso saber a disponibilidade de artigo(s) em armazém. Do armazém, importa saber o seu local. Quanto aos fornecedores, que abastecem os armazéns com roupa, é necessário saber o seu nome de empresa, o país de origem, o e-mail, o endereço e o telefone. A quantidade de roupa no armazém deve ser sempre atualizada (stock).

Estando a roupa disponível, é efetuado o pagamento da compra. Do pagamento, interessa saber a sua data, o montante, a fatura e, além disso, este tem de ser feito através de um método, seja ele por PayPal, Multibanco ou MB Way, por exemplo.

Realizada a encomenda (da qual é gerada um nº), é operado o envio. O envio tem como destino uma morada de envio e tem sempre uma data de entrega. Este é feito sempre por uma transportadora, de quem se importa saber o nome, o país de origem, o e-mail e o telefone. É importante que a transportadora atualize as informações sobre o estado de envio. Além disso, a encomenda pode ainda ser cancelada se for feita antes de ser procedido o seu envio. Estes cancelamentos têm um número de processo, um motivo, data de início, data de fim e um resultado atribuído pela loja.

Sendo assim, a pessoa que adquire esta base de dados pode realizar ou aceder a certos dados tais como o número de roupas que existem e as suas categorias, se a loja é frequentada por mais homens ou mulheres, quais os produtos mais adquiridos pelos seus clientes, entre outros.

B- Conceptual Modeling

C- Generative AI Integration

A AI utilizada pelo nosso grupo foi o Chat GPT.

Começámos por lhe pedir para fazer um UML sobre o nosso tema, tendo ele dado como resposta às seguintes: Cliente, Produto, Carrinho De Compras, Pedido, Pagamento, Stock, Categoria, Avaliação, Fornecedor, Funcionário, Cupão, Endereço, Log Atividades, Notificação e Transportadora. Constatámos que a maior parte das classes que a AI forneceu já estavam incluídas no nosso UML tivessem ou não o mesmo nome, à exceção de Pedido, Funcionário, Endereço, Log Atividades e Notificações.

Em relação à inclusão destas classes no nosso trabalho, não decidimos colocar/acrescentar por diversas razões. Em vez da classe Pedido, nós achámos melhor a colocação da entidade Encomenda, que tem o seu respectivo número. Uma vez que o nosso UML está relacionado com a gestão de uma loja de roupa online, não faria sentido adicionar uma classe Funcionário. O mais semelhante a isso seria falar sobre um possível Administrador do site, que faria manutenção do mesmo. Esta ideia foi debatida logo no início do projeto por todos os elementos do grupo, mas acabou por ser posta de parte, pois, o nosso projeto é mais focado na compra de um cliente e no restante processo de envio para o mesmo e não tanto na logística feita por trás de um site. Da mesma forma, não são colocadas as classes Log Atividades e Notificações. A classe Endereço não é uma classe no nosso projeto pois este ponto é representado pelo nosso atributo Morada De Envio, associado à classe Envio.

Em seguida, o Chat GPT concedeu às nossas classes os seus respectivos atributos, tendo eles sido os seguintes:

"Cliente:

Nome
Sobrenome
Endereço de e-mail
Senha
Número de telefone

Avaliação:

Texto da avaliação
Classificação (estrelas)
Data da avaliação

Compra:

Data da compra
Valor total

Status (processando, entregue, etc.)

Promoção:

Nome da promoção

Descrição

Data de início

Data de término

Desconto (percentagem ou valor)

Pagamento:

Método de pagamento (cartão de crédito, PayPal, etc.)

Número de cartão

Data de vencimento do cartão

Nome no cartão

Roupa:

Nome

Descrição

Preço

Tamanho

Cor

Marca

SKU (Unidade de Manutenção de Estoque)

Imagem do produto

Fornecedor:

Nome da empresa

Contato

Endereço

Número de telefone

E-mail

Armazém:

Localização

Capacidade de armazenamento

Status (ativo, inativo, etc.)

Transportadora:

Nome da empresa de transporte

Contato

Número de telefone

Métodos de envio oferecidos

Encomenda:

Número do pedido
Data do pedido
Status (pendente, entregue, etc.)

Envio:

Data de envio
Método de envio
Número de rastreamento

Devolução:

Data de devolução
Motivo da devolução
Status da devolução

Stock:

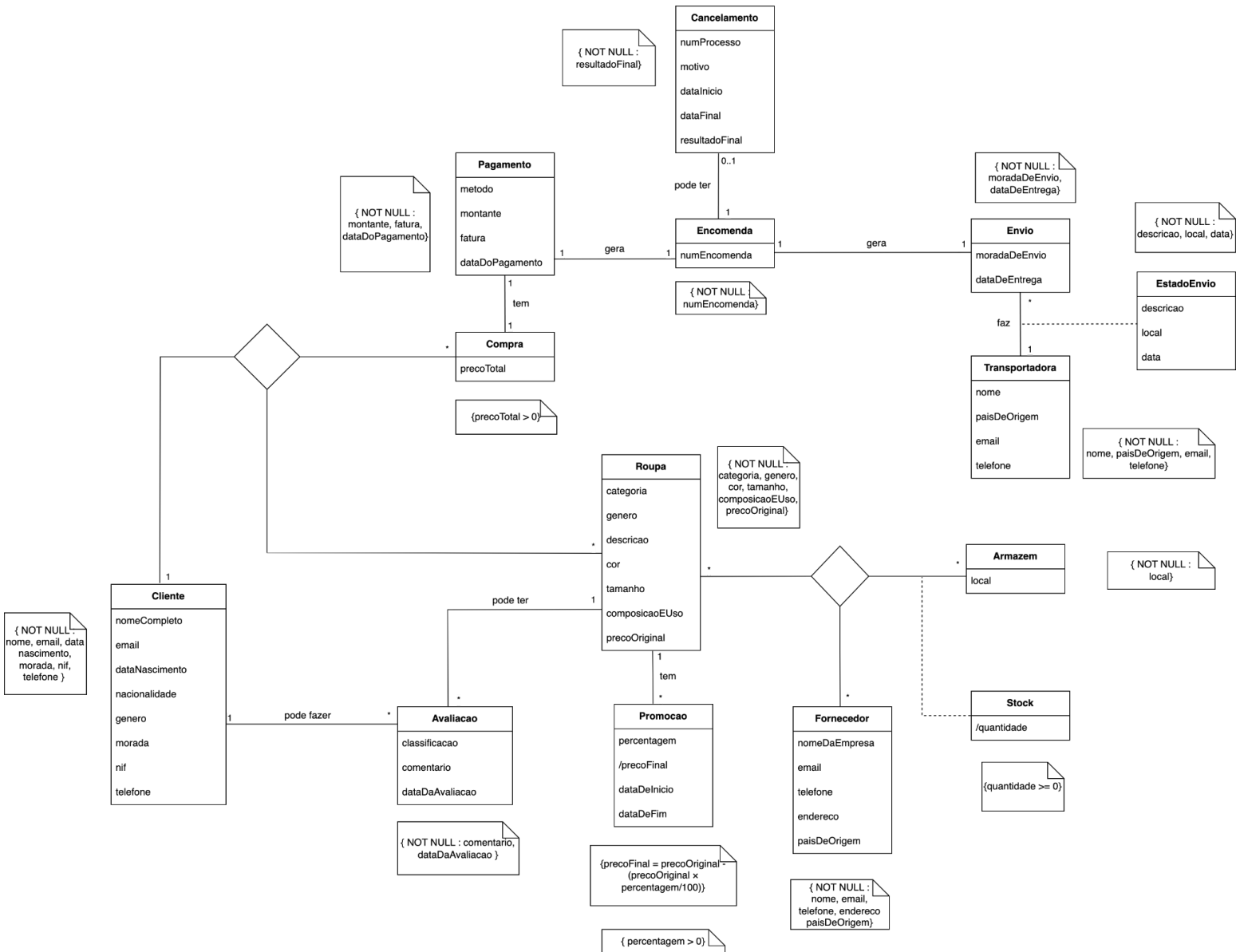
Quantidade disponível
Quantidade mínima de estoque
Quantidade máxima de estoque"

Nesta etapa, nós decidimos adicionar alguns atributos que não tínhamos presentes anteriormente. Adicionamos um número de telefone ao nosso Cliente, pois é importante existir um segundo meio de comunicação. Não tínhamos também um email e um número de telefone associados ao Fornecedor e à Transportadora, e por isso acrescentamos esses atributos a essas classes. A Data de Avaliação foi também um ponto que não nos tinha ocorrido antes, e por isso também a sua colocação. Além disso, achámos pertinente a colocação de datas de início e fim da Promoção, pois é uma informação relevante sobre uma determinada roupa.

Finalmente, pedimos à AI para nos mostrar que relações faria com as classes que escolhemos colocar no nosso UML. Neste passo, a inteligência artificial concebeu praticamente as mesmas relações que nós tínhamos previamente feito no nosso UML. Contudo, além das relações dadas pelo Chat GPT que o nosso trabalho já continha, ainda tínhamos mais relações que nós considerámos igualmente importantes.

SECOND SUBMISSION

5.1

A- Refine the conceptual model

B- Define the relational schema

Cliente(nif, nomeCompleto, email, dataNascimento, nacionalidade, genero, morada, telefone)

Roupa(idRoupa, tamanho, categoria, genero, descricao, cor, composicaoEUso, precoOriginal)

Pagamento(fatura, metodo, montante, dataDoPagamento, idCompra->Compra)

Encomenda(numEncomenda, fatura->Pagamento)

Promocao(idPromocao, percentagem, precoFinal, dataDeInicio, dataDeFim, idRoupa->Roupa)

Fornecedor(idFornecedor, nomeDaEmpresa, email, telefone, endereco, paisDeOrigem)

Armazem(idArmazem, local)

Transportadora(idTransportadora, nome, paisDeOrigem, email, telefone)

Envio(idEnvio, moradaDeEnvio, dataDeEntrega, numEncomenda->Encomenda)

Cancelamento(numProcesso, motivo, dataInicio, dataFinal, resultadoFinal, idEncomenda->Encomenda)

Compra(idCompra, precoTotal)

Avaliacao(idAvaliacao, classificacao, comentario, dataDaAvaliacao, nif->Cliente, idRoupa->Roupa, tamanho->Roupa)

EstadoDeEnvio(idEstado, idEnvio->Envio, idTransportadora->Transportadora, descricao, data, local)

TransportadoraArmazem(idTransportadora->Transportadora, idArmazem->Armazem)

Stock(idFornecedor->Fornecedor, idArmazem->Armazem, idRoupa->Roupa, tamanho->Roupa, quantidade)

HistoricoDeCompras(nif->Cliente, idCompra->Compra, idRoupa->Roupa, tamanho->Roupa)

Refinamentos feitos após a integração com a IA:

Fornecemos ao ChatGPT uma cópia no nosso modelo relacional, dando-lhe todos os dados necessário, perguntamos se haveria alguma alteração que pudesse ser feita e a resposta obtida pelo mesmo foi: "Seu esquema relacional parece estar bem estruturado, com chaves primárias e estrangeiras apropriadas, refletindo as relações entre as entidades do sistema. As chaves estrangeiras estão devidamente relacionadas e parecem seguir as regras estabelecidas para garantir integridade referencial. Com base no esquema e nas informações fornecidas, não identifiquei problemas críticos que exijam ajustes imediatos. No geral, parece uma representação sólida das relações no seu sistema."

C. Functional Dependencies and Normal Forms Analysis**Cliente:**

{nif} -> {nomeCompleto, email, dataNascimento, nacionalidade, genero, morada, telefone}

3rd Normal Form: SIM
Boyce-Codd Normal Form: SIM

Cancelamento:

{numProcesso} -> {motivo, dataInicio, dataFinal, resultadoFinal, idEncomenda}

3rd Normal Form: SIM
Boyce-Codd Normal Form: SIM

Roupa:

{idRoupa, tamanho} -> {categoria, genero, descricao, cor, composicaoEUso, precoOriginal}

3rd Normal Form: SIM
Boyce-Codd Normal Form: SIM

Compra:

{idCompra} -> {precoTotal}

3rd Normal Form: SIM
Boyce-Codd Normal Form: SIM

Pagamento:

{fatura} -> {metodo, montante, dataDoPagamento, idCompra}

3rd Normal Form: SIM
Boyce-Codd Normal Form: SIM

Avaliacao:

{idAvaliacao} -> {classificacao, comentario, dataDaAvaliacao, nif, idRoupa, tamanho}

3rd Normal Form: SIM
Boyce-Codd Normal Form: SIM

Encomenda:

{numEncomenda} -> {fatura}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Envio:

{idEnvio} -> {moradaDeEnvio,
dataDeEntrega, numEncomenda}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Promocao:

{idPromocao} -> {percentagem,
preçoFinal, dataDeInício, dataDeFim,
idRoupa}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

EstadoDeEnvio:

{idEstado} -> {idEnvio, idTransportadora,
descricao, data, local}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Fornecedor:

{idFornecedor} -> {nomeDaEmpresa,
email, telefone, endereco,
paisDeOrigem}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Stock:

{idFornecedor, idArmazem, idRoupa,
tamanho} -> {quantidade}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Armazem:

{idArmazem} -> {local}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

HistoricoDeCompras:

{idCompra, idRoupa, tamanho} -> {nif}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Transportadora:

{idTransportadora} -> {nome,
paisDeOrigem, email, telefone}

3rd Normal Form: SIM

Boyce-Codd Normal Form: SIM

Todas as relações anteriormente mencionadas estão em conformidade com a Terceira Forma Normal (3FN) e com a Forma Normal de Boyce-Codd (BCNF).

Analisando uma por uma:

-Cliente(nif, nomeCompleto, email, dataNascimento, nacionalidade, genero, morada, telefone)
{nif} -> {nomeCompleto, email, dataNascimento, nacionalidade, genero, morada, telefone}

Neste caso, não há qualquer dúvida sobre a existência de alguma violação pois, todos os atributos, nomeCompleto, email, dataNascimento, nacionalidade, genero, morada, telefone, têm total dependência no nif, por isso estão na Terceira Forma Normal e na Forma Normal de Boyce-Codd.

-Roupa(idRoupa, tamanho, categoria, genero, descricao, cor, composicaoEUso, precoOriginal)
{idRoupa, tamanho} -> {categoria, genero, cor, composicaoEUso, precoOriginal}

A relação está na terceira Forma Normal, assim como na Forma Normal Boyce-Codd. A chave primária composta idRoupa, tamanho identifica exclusivamente cada registro, e todos os outros atributos categoria, genero, cor, composicaoEUso, precoOriginal são totalmente dependentes dela. Ou seja, não há nenhuma violação.

-Pagamento(fatura, metodo, montante, dataDoPagamento, idCompra)
{fatura} -> {metodo, montante, dataDoPagamento, idCompra}

Tal como acontece em Cliente, nesta situação, a chave primária, representada por fatura, é capaz de identificar de forma única todos os atributos, tais como metodo, montante, dataDoPagamento, e idCompra, estes dependem inteiramente desta chave primária. Por isso, não são observadas nenhuma violações.

-Encomenda(numEncomenda)
{numEncomenda} -> {fatura}

A chave primária, composta apenas por numEncomenda, é suficiente para determinar o atributo da relação. Cada número de encomenda é único e identifica de maneira única os detalhes associados a essa encomenda em específico.

-Promocao(idPromocao, percentagem, precoFinal, dataDelnicio, dataDeFim, idRoupa)
{idPromocao} -> {percentagem, precoFinal, dataDelnicio, dataDeFim, idRoupa}

Não há dúvidas quanto à existência de violações, pois o idPromocao determina todos os atributos, portanto não há nenhuma violação.

-Fornecedor(idFornecedor, nomeDaEmpresa, email, telefone, endereco, paisDeOrigem)
{idFornecedor} -> {nomeDaEmpresa, email, telefone, endereco, paisDeOrigem}

Da mesma forma que aconteceu com a dependência funcional anterior, não há violações.

-Armazem(idArmazem, local)
{idArmazem} -> {local}

Neste caso, a chave primária idArmazem é capaz de identificar unicamente cada registo, e o atributo local é completamente dependente desta chave. Todos os atributos estão diretamente relacionados à chave primária. Assim, a relação não viola a Terceira Forma Normal nem a Forma Normal de Boyce-Codd.

-Transportadora(idTransportadora, nome, paisDeOrigem, email, telefone)
{idTransportadora} -> {nome, paisDeOrigem, email, telefone}

A relação está em conformidade com a Terceira Forma Normal (3FN) e a Forma Normal de Boyce-Codd (BCNF). Isto significa que não há violações nessas formas normais. A chave primária, idTransportadora, determina de maneira única todos os atributos, nome, paisDeOrigem, email, telefone.

-Envio(idEnvio, moradaDeEnvio, dataDeEntrega, numEncomenda)
{idEnvio} -> {moradaDeEnvio, dataDeEntrega, numEncomenda}

Não há dúvidas de que idEnvio determina moradaDeEnvio, dataDeEntrega e numEncomenda, por isso não há nenhuma violação.

-Cancelamento(numProcesso, motivo, dataInicio, dataFinal, resultadoFinal, idEncomenda -> Encomenda)
{numProcesso} -> {motivo, dataInicio, dataFinal, resultadoFinal, idEncomenda}

A mesma justificação aplica-se neste caso. Logo, também não há violações.

-Compra(idCompra, precoTotal)
{idCompra} -> {precoTotal}

Neste caso, idCompra determina o atributo precoTotal, por isso não há violação de nenhuma das formas.

-Avaliacao(idAvaliacao, classificacao, comentario, dataDaAvaliacao, nif->Cliente, idRoupa->Roupa, tamanho->Roupa)
{idAvaliacao} -> {classificacao, comentario, dataDaAvaliacao, nif, idRoupa, tamanho}

Esta relação está em conformidade com a Terceira Forma Normal (3FN) e a Forma Normal de Boyce-Codd (BCNF). Não há violações, pois a chave primária idAvaliacao determina todos os atributos relacionados.

-EstadoDeEnvio(idEstado, idEnvio->Envio, idTransportadora->Transportadora, descricao, data, local)
{idEstado} -> {idEnvio, idTransportadora, descricao, data, local}

Não há nenhuma violação. A chave primária idEstado determina os atributos idEnvio, idTransportadora, descricao, data e local.

-Stock(idFornecedor->Fornecedor, idArmazem->Armazem, idRoupa->Roupa, tamanho->Roupa, quantidade)
{idFornecedor, idArmazem, idRoupa, tamanho} -> {quantidade}

A relação está em conformidade com a Terceira Forma Normal (3FN) e a Forma Normal de Boyce-Codd (BCNF). Não há violações, pois a chave primária composta, formada por idFornecedor, idArmazem, idRoupa e tamanho, determina unicamente a quantidade.

-HistoricoDeCompras(nif->Cliente, idCompra->Compra, idRoupa->Roupa, tamanho->Roupa)
{idCompra, idRoupa, tamanho} -> {nif}

Uma vez que a chave composta, idCompra, IdRoupa e tamanho determina todos os atributos, podemos concluir que se encontra na 3FN E BCNF.

Refinamentos feitos após a integração com a IA:

Após termos perguntado à IA acerca do nosso modelo relacional, apresentamos-lhe também uma cópia das nossas dependências funcionais e perguntamos se concordava com a nossa resolução. A resposta que obtivemos foi um modelo relacional muito parecido ao nosso. A única diferença que apresentava era a seguinte:

“{nif} → {nomeCompleto, e-mail, dataNascimento, idade, nacionalidade, género, morada, telefone}
{e-mail} → {nif, nomeCompleto, dataNascimento, idade, nacionalidade, género, morada, telefone}”.

Contudo, decidimos não optar por essa forma pois achamos que não seria necessário.

Em seguida, perguntamos se havia alguma violação tanto da Terceira Forma Normal (3FN) como da Forma Normal de Boyce-Codd (BCNF), e a resposta dada pelo ChatGPT foi: “Com base nas dependências funcionais fornecidas, as relações parecem estar em conformidade com a Terceira Forma Normal (3FN) e a Forma Normal de Boyce-Codd (BCNF). Não foram identificadas violações evidentes nessas formas normais nas tabelas apresentadas. Cada relação tem uma chave primária identificada, e as dependências funcionais parecem estar adequadamente normalizadas. ”. Ou seja, acabamos também por não realizar nenhuma alteração.

D. SQLite Database Creation

create1.sql

Refinamentos feitos após a integração com a IA:

create2.sql

Demos todas as informações úteis à IA sobre o nosso Database Creation feito e em seguida perguntamos que alterações poderiam ser feitas e se havia algum erro. A resposta obtida foi que, no geral, o nosso SQL estava compreensivo e completo. No entanto, a IA deu-nos duas sugestões muito pertinentes de como a podíamos melhorar, as quais implementámos e passamos a indicar de seguida. Primeiro, chamou-nos a atenção para a consistência dos nomes das tabelas e atributos. Por exemplo, tínhamos chaves primárias a começar por “id” e outras a começar por “num”, quando ambas remetem para a mesma ideia. Corrigindo isso conseguimos melhorar a leitura e coerência do nosso código. Segundo, na primeira versão do nosso SQL, tínhamos optado por colocar a linha “DROP TABLE” antes da criação de cada tabela. Por sugestão da IA, argumentando que diminui a ocorrência de erros, decidimos colocar todas essas linhas no início do ficheiro, antes de qualquer criação de tabela.

D. Data Loading

populate1.sql

Refinamentos feitos após a integração com a IA:

populate2.sql

Nesta parte, solicitamos à IA para introduzir mais alguns dados na nossa base de dados, através da linha “INSERT INTO”. Por exemplo, no caso de uma Roupa, com os atributos (idRoupa, categoria, género, breveDescrição, tamanho, cor, composicaoEUso, preçoOriginal), demos um exemplo de uma entrada e algumas regras dos seus atributos, pedindo mais um número x de entradas.

6.1 - PARTICIPATION OF THE VARIOUS STUDENTS OF THE GROUP

Na primeira submissão, as tarefas do nosso grupo foram divididas de igual modo. Todos estiveram presentes na escolha do tema em aula, na seleção de classes e atributos fora da aula, tal como na elaboração do texto. Ao longo do tempo e com o aparecimento de novas ideias, fomos nos reunindo à distância para aprimoramento da base de dados. Os constraints foram também colocados por todos ao mesmo tempo. Por fim, reunimo-nos uma última vez para realizar a última etapa desta primeira submissão que consistia na elaboração de um UML pela AI. Aí tiramos as nossas conclusões e finalizamos o UML e os textos.

Por fim, na segunda submissão, optamos por utilizar a mesma estratégia, distribuindo várias tarefas por cada elemento e fomos reunindo de forma a analisar a proposta de cada um.