**Uma imagem com Gráficos, Tipo de letra, captura de ecrã, design

Descrição gerada automaticamente**

**Franquia de Lojas de Penhores**

**Base de Dados**

**Licenciatura em Engenharia Informática e Computação**

**Docentes:**

Prof. Carla Lopes, [ctl@fe.up.pt](mailto:ctl@fe.up.pt)

Prof. Michel Ferreira, [mpferrei@fc.up.pt](mailto:mpferrei@fc.up.pt)

Prof. Mariana Dias, [mfdias@fe.up.pt](mailto:mfdias@fe.up.pt)

**Discentes:**

Carolina Mosqueiro, [up202303637@fe.up.pt](mailto:up202303637@fe.up.pt)

Mafalda Dias, [up202307365@fe.up.pt](mailto:up202307365@fe.up.pt)

Maria Catarina Moreira, [up202306603@fe.up.pt](mailto:up202306603@fe.up.pt)

Turma 5, Grupo 2

Porto, 20 de outubro de 2024

**Índice**

[Descrição do Tema 3](#_Toc180250315)

[Modelo Inicial 3](#_Toc180250316)

[Descrição 3](#_Toc180250317)

[Modelo Conceptual 5](#_Toc180250318)

[Integração de IA 6](#_Toc180250319)

[Modelo Final 8](#_Toc180250320)

[Descrição 8](#_Toc180250321)

[Modelo Conceptual 9](#_Toc180250322)

# **Descrição do Tema**

A Base de Dados expressa neste relatório pelos membros do grupo 502 pretende simular um sistema de franchising aplicado ao típico funcionamento de lojas de penhores. Como tal, todos os estabelecimentos associados a esta franquia estão estruturados de maneira similar e, no que diz respeito ao seu funcionamento, seguem um conjunto fixo de regras e padrões, nomeadamente em relação aos empréstimos que fornecem aos seus clientes, às taxas de juros associadas e aos tipos de artigos que podem penhorar.

No que diz respeito à definição básica de lojas de penhores, estas são estabelecimentos que não só vendem certos tipos de artigos – tais como eletrodomésticos, joias, antiguidades, entre outros –, mas também os adquirem de clientes que, em troca, recebem um empréstimo – geralmente em dinheiro e que não lhes foi possível obter através de meios tradicionais. Estes artigos penhorados não ficam imediatamente na posse da loja, pois é concedido um intervalo de tempo aos clientes para devolverem o valor emprestado – agora com juros adicionados – de forma a recuperarem os seus itens. Caso não consigam fazê-lo até ao fim do prazo outrora estabelecido, estes passam a pertencer à loja e podem ser vendidos normalmente a entidades interessadas. Os detalhes inerentes aos itens são estabelecidos pelos funcionários das lojas.

O sistema de franchising aplicado a este conceito tem, como vantagem, a partilha de dados entre as diversas lojas associadas, particularmente no que diz respeito aos seus clientes e artigos.

# **Modelo Inicial**

## **Descrição**

**Loja:** A classe "Loja" pretende representar os diversos estabelecimentos pertencentes à franquia. Todos possuem um ID, nome, endereço e contacto telefónico. Cada loja tem dois inventários, da mesma forma que cada um desses inventários só pertence a uma loja.

**Inventário:** A classe “Inventário” procura distinguir os itens penhorados daqueles que se encontram à venda. Como tal, define todos os artigos da loja, os quais, pelo contrário, recebem um estado através desta classe.

**Item:** A classe “Item” define quer os artigos penhorados quer aqueles a serem vendidos. Estes possuem um ID, um estado e a data em que se integraram na loja. Um item pode ou não estar à venda e pode ou não estar penhorado, que, por conseguinte, se transparece num empréstimo. Cada artigo só pertence a um cliente, mas um cliente pode penhorar mais do que um artigo. Cada item possui uma descrição.

**Descrição:** A classe “Descrição” está diretamente associada a um item, na medida em que inclui o seu nome, a categoria a que pertence, as suas dimensões, o material de que é feito e o seu valor estimado. Está incluído, ainda, um ID da descrição.

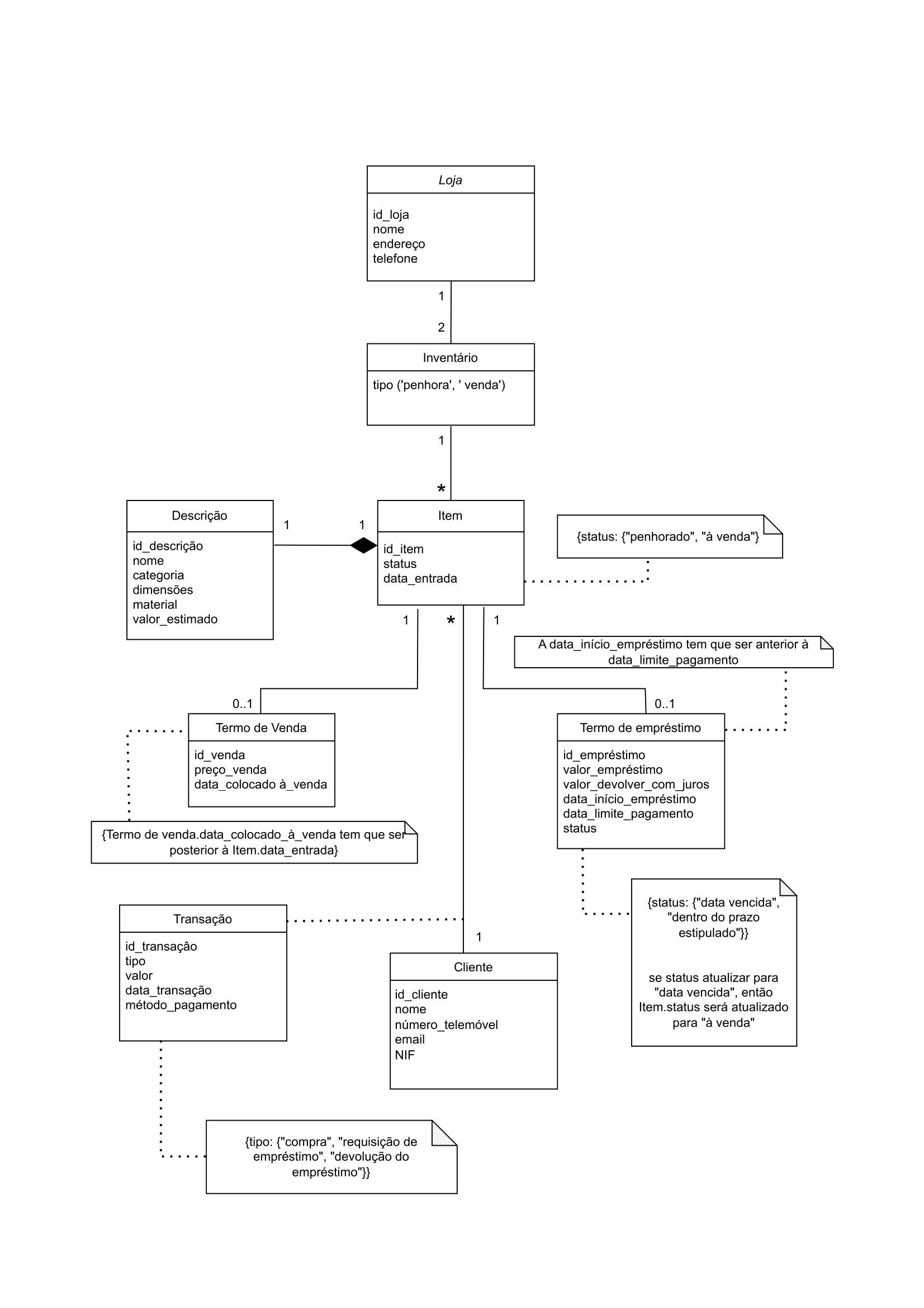
**Termo de Empréstimo:** A classe “Termo de Empréstimo” diz respeito ao empréstimo fornecido ao cliente de acordo com os itens penhorados. Inclui um ID, a quantia do empréstimo, os juros associados, a data em que foi realizado, a data limite para a retorna do empréstimo e respetivos juros e um estado, o qual define se o prazo para o seu pagamento foi ultrapassado ou não.

**Termo de Venda:** A classe “Termo de Venda” traduz a venda de um artigo a um cliente. Possui, como atributos, um ID, o preço do artigo e a data em que este foi vendido.

**Cliente:** A classe “Cliente” representa aquele que pretende penhorar ou comprar um item. Inclui um ID associado à pessoa, o seu nome, contacto telefónico, e-mail e NIF. Um cliente e o(s) seu(s) respetivo(s) artigo(s) desencadeiam uma transação.

**Transação:** A classe “Transação” simboliza, de um modo geral, o que o cliente pretendeu com a sua visita à loja: comprar um artigo, requisitar um empréstimo ou devolver a quantia outrora requisitada. Inclui um ID associado à transação, o seu tipo, valor transacionado, data da ocorrência e o método de pagamento utilizado.

## **Modelo Conceptual**



# **Integração de IA**

Escolheu-se, entre os vários modelos LLM, usar Claude.ai (versão 3.5 Sonnet). DESENVOLVER MAIS ISTO

Antes de abordar a IA, foram definidas algumas falhas do modelo inicial que se pretendia corrigir:

* Desenvolver mais o modelo de modo a que a sua complexidade corresponda integralmente à do tópico escolhido;
* Simplificar a distinção entre itens penhorados e itens à venda.

Decidimos começar com uma abordagem mais geral. Quando confrontado com a pergunta “O esquema UML anexado retrata uma possível base de dados para um franchising de lojas de penhores. Os clientes podem dirigir-se à loja para penhorar ou comprar itens. Os itens à venda são itens penhorados cujos empréstimos não foram saldados no prazo estipulado. O que mudarias neste esquema?”, o texto gerado pelo Claude.ai sugeriu várias alterações ao esquema, entre elas:

* O “rastreamento de funcionários” através da criação de uma nova classe denominada "Funcionário”, na qual seriam incluídos “atributos como id\_funcionario, nome, cargo, etc.”, sendo agora possível “associar cada transação ao funcionário que a realizou”;
* A “avaliação de itens”. A nova classe “Avaliação” (que inclui “atributos como id\_avaliacao […], valor\_estimado, data\_avaliacao”, etc.) permite criar uma ligação entre cada item e a sua avaliação formal, realizada por um funcionário;
* Melhorar a entidade “Descrição” ao “integrar os atributos da Descrição (nome, categoria, dimensões e material) diretamente na entidade Item”.

Revendo estas sugestões, decidiu-se aplicar as primeiras duas no modelo final. De resto, após alguma consideração, optou-se por descartar as mudanças propostas, já que os atributos relacionados à descrição do item, tal como a sua avaliação, são mais fáceis de interpretar como classes em separado. Adicionalmente, de forma a simplificar o esquema, decidiu-se tornar “Transação” numa classe de associação entre “Cliente” e “Item”.

Quando abordado com a questão “Quanto às relações entre ‘Loja’, ‘Inventário’ e ‘Item’: o nosso objetivo era criar uma divisão entre os itens para venda e os itens penhorados, no entanto, a classe ‘Item’ já tem um atributo ‘status’ com um efeito parecido. Como simplificarias/modificavas estas relações?”, o Claude.ai reconheceu a “redundância entre a entidade ‘Inventário’ e o atributo ‘status’ na entidade ‘Item’” e sugeriu uma simplificação.

As modificações resumem-se à remoção da classe “Inventário”, ligando diretamente “Loja” e “Item”, já que a sua função principal – distinguir entre itens para venda e itens penhorados – já está a ser cumprida pelo atributo “status” na classe “Item”.

Quanto à questão “Devido à sua importância, fará sentido incluir o ‘status’ do item como uma classe separada do esquema UML para facilitar a sua interpretação?”, a resposta do Claude.ai foi uma lista de prós e contras relativos a cada abordagem (manter “status” como um atributo de “Item” ou criar uma classe separada para “status”):

* A primeira abordagem resulta num “esquema mais simples”, mas também resulta numa menor visibilidade do “status”, tornando o modelo menos flexível, já que mudanças neste atributo “requerem alterações na estrutura da tabela”;
* A segunda opção “facilita a adição de novos status ou atributos relacionados ao status” e “destaca a importância do status no modelo”, permitindo também “rastrear facilmente mudanças de status ao longo do tempo”. No entanto, adiciona uma classe extra, o que resulta num modelo mais complexo.

Ponderando agora as mudanças sugeridas, decidiu-se remover a classe “Inventário”, mas também descartar a primeira proposta que mantinha “status” em “Item”. Como o estado (“à venda”/“penhorado”) é dinâmico, ou seja, muda após um dado número de dias – após “data vencida” –, achou-se pertinente retratar esta característica numa classe separada, denominada “Estado”, devido, principalmente, a duas vantagens: destacar a importância do estado e permitir a fácil adição de novos status sem alterar a estrutura da tabela “Item”.

# **Modelo Final**

## **Descrição**

As mudanças ao modelo podem ser resumidas de forma breve:

* Adicionaram-se duas novas classes, “Funcionário” e “Avaliação”, sendo que cada item passa a ter uma avaliação feita por um funcionário, em que é avaliado o estado de conservação do item e é estimado o seu valor;
* Criou-se também uma classe “Estado” após a remoção da classe “Inventário”, que visa mostrar, de forma clara, se o item foi penhorado e está armazenado até que seja pago o empréstimo a ele associado ou se foi retido pela loja e está disponível para venda.
* [[[[OK, ISTO TEM QUE SER VISTO:  
  TALVEZ FAÇA MAIS SENTIDO MUDAR PARA CLASSE DE ASSOCIAÇÃO NO RAFADO TB???  
  E assim tira-se este parágrafo. Mas tudo depende se acharem que foi pouca coisa alterada de uma fase para a outra idk]]]]

## **Modelo Conceptual**

