



UNIVERSIDADE  
AUTÓNOMA  
DE LISBOA

# Gestão logística de uma empresa de transportes

Licenciatura em Engenharia Informática  
Disciplina de Programação Orientada a Objetos

**Por:**

Pedro Martins – 30003748

Luís Martins – 30003039

Kiala Mawete – 30001234

Vítor Igrejas – 30000102

## Índice

Resumo .....	3
Diagrama de Classes (UML) .....	4
Desenvolvimento .....	5

## Resumo

Neste semestre encontramos-nos a desenvolver conceitos adjacentes à cadeira de programação orientada a objetos dentro das mais recentes boas práticas de desenvolvimento endereçadas em aula.

Devemos ter em conta todo o processo de planeamento prévio a uma implementação sólida do nosso programa. Este processo consiste no levantamento de requisitos e a análise dos mesmos de forma a transcrevê-los para um diagrama de classes que agilizará a criação ou implementação das mesmas.

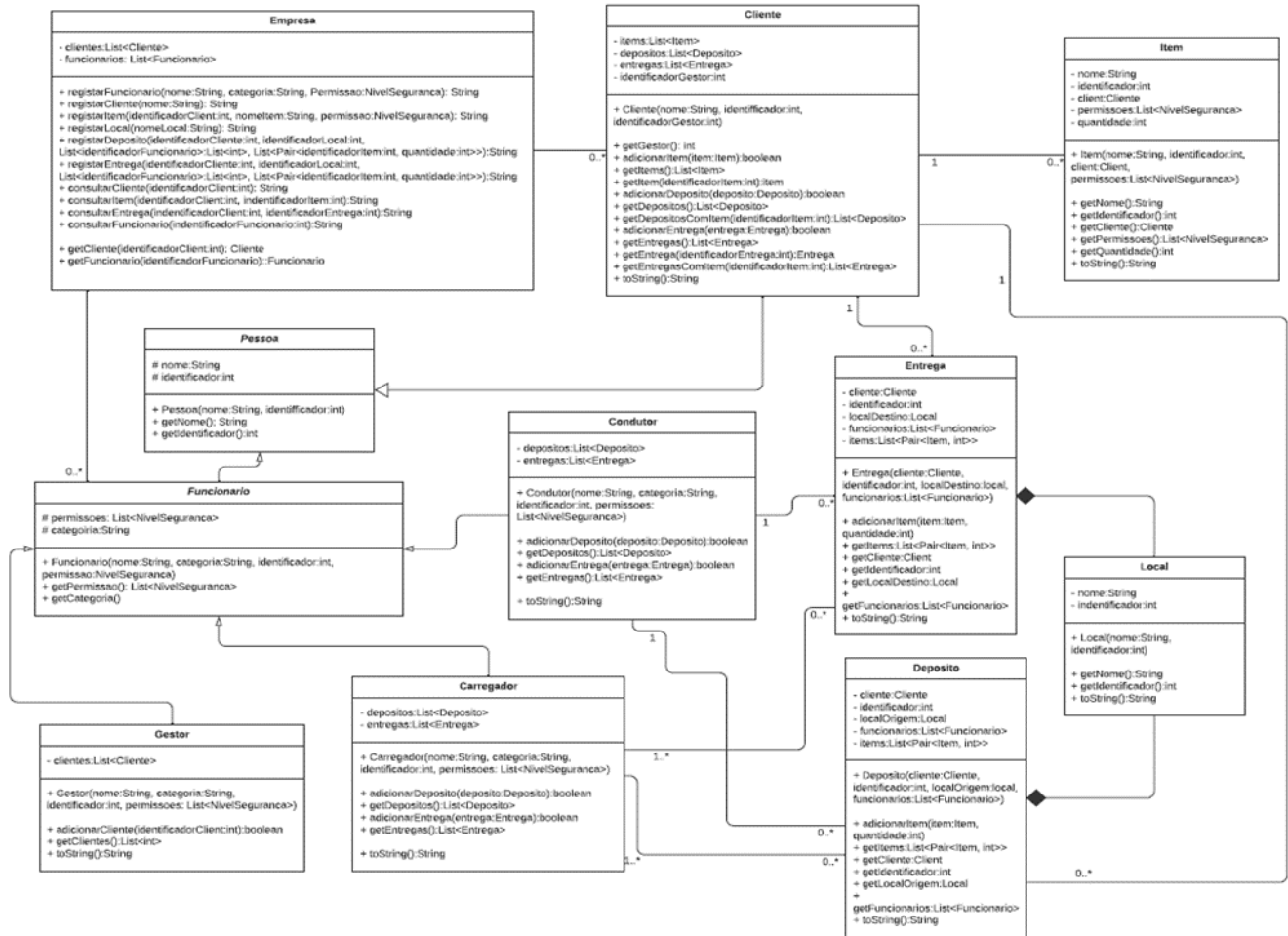
O tema abordado neste projeto consiste na criação de um sistema de gestão logística de uma empresa de transportes.

Iremos gerir toda a comitiva de profissionais e clientes, dado ser uma empresa contratada por clientes para efetuar o transporte de vários tipos de cargas teremos também a gestão destas mesmas cargas por cliente, assim como o depósito destes produtos e a entrega do mesmo.

No final devemos conseguir correr todos os comandos definidos em enunciado e a respetiva campanha de testes entregue pelo professor.

Na próxima página vamos poder observar a estrutura de classes definidas no âmbito deste projeto.

## Diagrama de Classes (UML)



## Desenvolvimento

O programa é responsável por processar todos os comandos disponíveis ao utilizador. Ao criar uma nova classe `TransportationUnit` e a disponibilizar a mesma a todos os comandos, podemos assim, invocar todos os métodos abstraídos na classe `TransportationUnit` da interface `TransportationUnitInterface`.

Todas as classes expõem os métodos refletidos em cada uma das suas interfaces criadas para essa mesma abstração.

Foram criadas ainda classes do tipo enumerados para os tipos de categoria e para o tipo de permissão. Estes mesmos enumerados são usados em diversas validações das strings inseridas pelo utilizador em cada um dos comandos em que isso seja necessário.

A nossa classe principal tem como objetivo expor todos os métodos que nos permitem:

- **Registar Funcionários**
  - Após validações requeridas em enunciado, será criada uma instância nova da classe funcionário e adicionada a lista de funcionários existentes da classe unidade de transporte, cada funcionário terá um id específico atribuído através de um incremento de ordem crescente sobre o valor 0.
- **Registar Clientes**
  - Tal como os funcionários o mesmo acontece aos clientes para uma lista própria criada para os mesmos, cada cliente terá um id específico atribuído através de um incremento de ordem crescente sobre o valor 0.
- **Registar Itens**
  - Introdução de dados referentes a um novo item, estes novos itens serão adicionados a uma lista de itens existente na nossa unidade de transporte onde irá haver a lista total de itens registados. No entanto, é nos pedido que os mesmos sejam salvaguardados para o cliente específico tendo os mesmos um id para cada cliente referido, isto é, a lista de itens será também ela guardada para cada cliente.
- **Registar Depósitos**
  - Introdução de depósitos de itens, onde é recebido valores de itens e quantidades para que sejam salvaguardados. Sendo os depósitos associados a cada cliente para itens por eles registados, estes ficam associados ao cliente onde é feita a introdução dos depósitos.
- **Registar Entregas**
  - Tal como nos depósitos o conceito de registo de entregas é o mesmo, sendo neste caso retirado a quantidade necessária a entrega dos itens existentes.
- **Consultas**
  - No programa somos capazes de consultar funcionários, clientes, itens e entregas. Estas consultas são ordenadas conforme os requisitos através do uso de métodos de consulta criados na unidade de transporte e a criação dos comparadores necessários.

- Gravar
  - Podemos serializar a classe de unidade de saúde para um ficheiro binário através dos métodos de serialização dados em aula, este método recebe o nome do ficheiro com o qual quer ser gravado.
- Ler
  - Através dos métodos estudados no ponto anterior conseguimos ler o ficheiro gravado com um nome igualmente atribuído no pedido de leitura de ficheiro, ao ler o ficheiro este reabre os objetos anteriormente gravados para prosseguir com a introdução de dados.