

Empresa de Transporte de Mercadorias (Parte 1)

Transportes SML

up201806429@fe.up.pt

Diogo Miguel Ferreira Rodrigues

up201806554@fe.up.pt

Telmo Alexandre Espirito Santo Baptista

up201306340@fe.up.pt

Luís Paulo da Rocha Miranda

Faculdade de Engenharia da Universidade do Porto
Algoritmos e Estruturas de Dados (AEDA) - Turma 5, grupo 3

16 de novembro de 2019

Problema: Empresa de Transporte de Mercadorias

Modelar um problema recorrendo ao paradigma da orientação por objetos e usar a linguagem C++ para implementar a solução correspondente.

A empresa Transportes SML é especialista no transporte de mercadorias. A empresa possui um número fixo de camiões de diferentes tipos, específicos para o transporte de determinada mercadoria [...]

Os clientes podem requisitar serviços de transporte de mercadorias, o que pode obrigar ao uso de múltiplos camiões por parte da empresa.

Interessa conhecer os valores mensais que a empresa retira dos seus serviços de transporte, para cada um dos tipos de camião e no geral. Deve ser possível monitorar os clientes da empresa e serviços efetuados.

Os utilizadores da aplicação devem fazer login, e possuem permissões diferentes consoante o seu tipo:

- Funcionário (Employee)
 - Gestor (Manager)
 - Condutor (Driver)
- Cliente (Client)

Um só tipo de camiões (Truck), que pode carregar uma carga (Cargo) de um dos tipos, cujo preço de transporte varia:

- Normal (Normal)
- Animal (Animal)
- Refrigerada (Refrigerated)
- Perigoso (Dangerous)

Todos os utilizadores podem ver e editar as suas informações pessoais. O cliente pode ver e editar os seus serviços ativos, ou requisitar um novo. O condutor pode ver e editar os seus serviços ativos, bem como ver a lista de camiões disponíveis e as suas estatísticas mensais.

O gestor pode fazer tudo isto e ainda ver as listas de clientes e funcionários (completas ou filtradas por um argumento específico), com possibilidade de edição para qualquer um destes, além de poder ver os valores mensais de toda a empresa e outras estatísticas.

Quando um cliente requisita um serviço, são atribuídos automaticamente condutores e camiões. Foram utilizados os algoritmos de ordenação mergesort e pesquisa linear para este efeito.

Diagrama de classes

- diagrama de classes - diagrama de classes

Estrutura de ficheiros

clients.txt

Guarda a lista de todos os usuários no seguinte formato: Nome; Número de telemóvel; Nome de utilizador; Password; Morada (Rua e número; Código-Postal; Distrito e País); NIF;

drivers.txt/managers.txt

Guarda a lista de todos os condutores no mesmo formato dos clientes, com uma linha extra equivalente ao salário. trucks.txt

Guarda a lista de todos os camiões da empresa no seguinte formato: Matrícula; Data da matrícula; Combustível (de 0 a 6, de acordo com o tipo); Alcance (em km); Categoria; Número de cargas; Lista de cargas (Peso; Descrição; Preço por km; custo de refrigeração ou nível de perigo, caso se aplique);

services.txt

Guarda a lista dos serviços no seguinte formato: ID; nome de utilizador (cliente); nome e número de telemóvel do contacto 1; nome e número de telefone do contacto 2; hora de início; hora de fim; distância; tipo de carga; peso da carga; descrição da carga; número de condutores alocados

Tratamento de exceções

Lista das exceções + breve explicação de cada uma

Lista de funcionalidades

Os clientes podem requisitar serviços, com origem, destino, duração da viagem, ...

Mostrar lista de clientes e funcionários; adicionar, editar ou remover clientes e funcionários;

Mostrar lists de camiões; adicionar, editar ou remover camiões;

Mostrar lista de serviços; adicionar, editar ou remover um serviço;

Login para clientes, gestores e condutores;

Quando um condutor chega ao destino, tem de ter um contacto que pode, ou nao, ser do cliente.

Os gestores podem ver valores mensais de lucro de um camião específico, um tipo de camiões, ou totais;

AS VÁRIAS LISTAS E OPÇÕES DAS LISTAS (ORDENAÇÃO, FILTRAGEM,...)

Destaque de funcionalidade

Trabalho de grupo = colaboração + trabalho individual

Neste trabalho as classes estavam muito interligadas umas com as outras e com a estrutura da aplicação, pelo que foi difícil compartimentar tarefas e classes/funções

Diogo Rodrigues

- utils: quase tudo, incluindo
stringregex,
mergesort, findif
- Person e todas as classes derivadas
- Truck: reescreveu
- Cargo: reescreveu
- I/O ficheiros
- App: tabelas, opções das tabelas e mostrar detalhes de entidades
- Time, Address, Phonenummer, Temperature, VAT
- Service e função de alocação

Telmo Baptista

- utils: algumas funções
- App: input do utilizador; adição, edição e remoção de entidades
- Cargo: implementou
- Person: alguns métodos
- Time: algumas funções
- Truck: implementou

Luís Miranda

- Organização da app
- Permissões
- Dados
- Testes
- Apresentação