



Universidade do Minho

Programação Orientada a Objetos

MIEI - 2º ANO - 2º SEMESTRE

UNIVERSIDADE DO MINHO

UMeR

Grupo 29

Rui Vieira

A74658



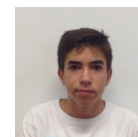
Frederico Pinto

A73639



José Sousa

A74678



3 de Junho de 2017

Conteúdo

1	Introdução	2
2	Descrição da aplicação	3
3	Arquitetura da aplicação	6
3.1	UMERApp	6
3.1.1	Memu	7
3.2	UMER	7
3.3	Localização	7
3.4	Utilizador	8
3.4.1	CLiente	8
3.4.2	Motorista	8
3.4.3	MotoristaE	9
3.5	Empresa	9
3.6	Viatura	9
3.6.1	Moto	10
3.6.2	Carro	10
3.6.3	Carrinha	10
3.7	Viagem	10
3.8	CatUtilizadores	11
3.9	CatViaturas	11
4	Perspectivas de melhorias e trabalho futuro	12
5	Conclusão	13

1. *Introdução*

Este projeto foi-nos solicitado pelos docentes da UC *Programação Orientada a Objetos*, onde nos foi proposta a realização de uma aplicação capaz de dar suporte a toda a funcionalidade que permita que um utilizador realize a viagem num táxi na empresa UMeR. Existem dois tipos de atores nesta aplicação, Cliente e Motorista. Os Clientes podem requisitar uma viagem, avaliar o motorista e consultar o seu histórico de viagens. Já os Motoristas só podem fazer a ultima do ator Cliente no entanto têm a possibilidade de adicionar um novo táxi, associar a si um táxi, registar uma viagem, sinalizar disponibilidade e associar-se a uma empresa tendo desta forma acesso á listagem de motoristas e táxis dessa mesma empresa podendo também desassociar-se da respetiva empresa.

A realização deste projeto tinha como principal objetivo implementar em java os conhecimentos adquiridos na respetiva UC, em enfase para *modularidade* e *encapsulamento de dados* que até então não tinham sido abordadas ao longo nosso percurso académico.

2. Descrição da aplicação

Esta é uma aplicação com uma interface muito simples, de forma a que o utilizador possa tirar o maior proveito. Todos os menus funcionam à base de opções por números. Quando a aplicação é iniciada, o primeiro menu a que está sujeito o utilizador é o seguinte:

```
***** Menu *****
1 - Registrar Utilizador
2 - Iniciar Sessao
3 - Lista de Empresas
4 - Top 10 Clientes
5 - Top 5 Motoristas
0 - Sair
*****
```

Neste menu, menu inicial, o utilizador pode registar-se, (Opção 1), e caso já o tenha feito pode iniciar sessão na Opção 2. Para além disso, ao escolher a Opção 3, o utilizador tem acesso à lista de Empresas existentes na aplicação. Ao escolher a Opção 4 ou Opção 5, o utilizador tem acesso ao Top 10 - Clientes que mais gastam e ao Top 5 - motoristas que apresentam mais desvios entre os valores previstos para as viagens e o valor final faturado.

Se o utilizador decidir efetuar o registo será apresentado o seguinte menu:

```
***** Menu *****
1 - Motorista
2 - Cliente
0 - Sair
*****
Opção:
```

Após a apresentação deste menu, o utilizador pode escolher em registar-se como Motorista (Opção 1) ou Cliente (Opção2). Para isso terá de inserir determinadas informações necessárias para o registo:

Nome: José Alves
Email: alves@gmail.com
Password: praia
Morada: Portimão
Data de nascimento: 21/05/78

Depois do registo, o utilizador pode agora iniciar sessão e desfrutar da aplicação. Depois de iniciar sessão será apresentado um menu consoante o tipo de utilizador que é, caso seja um Motorista, o seguinte menu é apresentado:

```
***** Menu *****  
1 - Adicionar Veiculo  
2 - Associar Veiculo  
3 - Associar Empresa  
4 - Consultar Historico  
5 - Sinalizar Disponibilidade  
6 - Terminar Sessão  
0 - Sair  
*****  
Opção:
```

Opção 1 – Esta opção permite ao utilizador (Motorista), registar uma viatura na aplicação.

Opção 2 – Permite ao utilizador associar uma viatura a si próprio.

Opção 3 - Permite ao utilizador associar-se a uma empresa, caso ela exista.

Opção 4 - Com esta opção o utilizador pode consultar o seu histórico de viagens.

Opção 5 - Nesta opção o utilizador (Motorista) pode mudar a sua disponibilidade.

Opção 6 - O utilizador termina sessão na aplicação.

Opção 0 - Volta atrás no menu.

Caso o Motorista, escolha a Opção 3 e esta seja efetuada com sucesso, o Motorista passará a ser um Motorista de uma empresa e terá acesso a outro menu diferente:

Opção 1 (Adicionar Veiculo) – Esta opção permite ao utilizador (Motorista), registar uma viatura na aplicação.

Opção 2 (Associar Veiculo) - Permite ao utilizador associar uma viatura a si próprio.

Opção 3 (Consultar Histórico) - Com esta opção o utilizador pode consultar o seu histórico de viagens.

Opção 4 (Lista de Motoristas da Empresa) – Nesta opção é apresentada a lista de motoristas associados à empresa.

Opção 5 (Lista de Viaturas duma Empresa) – Nesta opção é apresentada a lista de viaturas associadas à empresa.

Opção 6 (Sinalizar Disponibilidade) - Nesta opção o utilizador (Motorista) pode mudar a sua disponibilidade.

Opção 7 (Desassociar Empresa) – Nesta opção o utilizador desassocia-se da empresa a que está ligado.

Opção 8 (Terminar Sessão) – O utilizador termina sessão na aplicação.

Opção 0 - Volta atrás no menu.

No entanto, caso o utilizador seja um Cliente o menu apresentado é o seguinte:

```
***** Menu *****
  1 - Avaliar Motorista
  2 - Consultar Historico
  3 - Solicitar Viagem
  4 - Terminar Sessão
  0 - Sair
*****
Opção:
```

Opção 1 – Nesta opção o Cliente pode avaliar o Motorista após a viagem.

Opção 2 – Com esta opção o utilizador pode consultar o seu histórico de viagens.

Opção 3 – Nesta opção o utilizador solicita uma viagem.

Opção 4 - O utilizador termina sessão na aplicação.

Opção 0 - Volta atrás no menu.

Caso o Cliente escolha a Opção 3 (Solicitar Viagem), será apresentado no ecrã o seguinte menu:

```
***** Menu *****
  1 - Registrar Utilizador
  2 - Iniciar Sessao
  3 - Lista de Empresas
  4 - Top 10 Clientes
  5 - Top 5 Motoristas
  0 - Sair
*****
```

Opção 1 - Caso o utilizador escolha esta opção solicita o táxi mais próximo dele.

Opção 2 – Nesta opção o cliente pode solicitar um táxi específico podendo escolher entre carro, moto ou carrinha.

3. *Arquitetura da aplicação*

3.1 UMERApp

Atributos:

- UMER um
Umer a correr na aplicação
- Menu_principal
Menu principal.
- Menu_menu_registo
Menu de registo.
- Menu_menu_motorista
Menu para motoristas.
- Menu_menu_motoristaEmp
Menu para motoristas de empresa.
- Menu_menu_cliente
Menu para clientes.
- Menu_menu_solicitar
Menu para solicitar uma viagem.
- Menu_menu_veiculo
Menu para os tipos de veiculo

Esta classe trata de toda a interface apresentada ao utilizador, para além disso esta classe trata de gravar o estado da aplicação e ler esse mesmo quando a aplicação é fechada e reiniciada.

3.1.1 Menu

Atributos

- List<String> opcoes
Lista de opções que o menu contém.
- int op
Operação escolhida pelo utilizador.

Esta classe é a classe responsável por todos os menus criados e pelo funcionamento dos mesmos. Com esta classe o trabalho da classe principal, UMERApp, é muito mais fácil.

3.2 UMER

Atributos

- Utilizador uConectado;
Utilizador que está conetado na aplicação. (com sessão iniciada).
- CatUtilizadores catU;
CatUtilizadores(TreeSet<Utilizadores>) que contém todos os utilizadores registados na Umer.
- CatViaturas catV;
CatViaturas(TreeSet<Viatura>) que contém todas as viaturas registadas na Umer.
- ArrayList<Empresa> catE;
Lista que contém todas as empresas existentes na Umer.

Esta classe é responsável pelo funcionamento da aplicação, é na própria que estão desenvolvidas praticamente todas as funcionalidades da nossa aplicação.

3.3 Localização

Atributos

- int x
Coordenada x.
- int y
Coordenada y.

A classe Localização é responsável pela criação de uma localização (x,y) e dos cálculos de distâncias.

3.4 Utilizador

Atributos

- String email
Email do Utilizador.
- String nome
Nome do Utilizador.
- String password
Password do utilizador.
- String morada
Morada do Utilizador.
- String dataNascimento
Data de nascimento do Utilizador.
- Localizacao local
Localização do Utilizador.
- TreeSet<Viagem> catViagens
Coleção de viagens efetuadas pelo Utilizador.

Classe abstrata para um utilizador, todos os utilizadores da aplicação têm em comum o que se encontra desenvolvido na própria.

3.4.1 CLiente

Esta classe corresponde a um Cliente, esta classe herda todos os parâmetros da classe anterior.

3.4.2 Motorista

Atributos

- int pontualidade
Pontualidade do motorista.
- int classificação
Classificação do motorista.
- long kms
Kilómetros totais percorridos pelo motorista.
- boolean disponível
Disponibilidade do motorista.

- ArrayList<Integer> aval

Lista com as avaliações do motorista feita pelos clientes.

Esta classe corresponde a um Motorista, e tal como a classe Cliente, esta classe herda os atributos da classe Utilizador aos quais são adicionados os atributos acima descritos.

3.4.3 MotoristaE

Atributos

- Empresa empresa

Empresa à qual o motorista pertence.

- boolean disp

Disponibilidade do motorista.

Esta classe representa um motorista de empresa. Esta classe só tem métodos mais comuns.

3.5 Empresa

Atributos

- long id

Id da Empresa.

- CatUtilizadores motoristas

TreeSet<Utilizadores> com todos os motoristas da empresa.

- CatViaturas viaturas

TreeSet<Viaturas> com todas as viaturas da empresa.

Esta classe corresponde a uma Empresa. Esta classe só tem métodos mais comuns.

3.6 Viatura

Atributos

- String matricula

Matricula da viatura.

- Motorista motorista

Motorista da viatura.

- double velMedia

Velocidade média da viatura.

- double custo
Preço por Km da viatura.
- double fiabilidade
Fiabilidade da viatura.
- boolean disp
Disponibilidade da viatura.
- Localizacao local;
Localização da viatura.
- TreeSet<Viagem> catViagens;
Coleção de todas as viagens efetuadas pela viatura.

Esta classe corresponde a uma Viatura. Todas as viaturas têm em comum os atributos presentes nesta classe.

3.6.1 Moto

Esta classe corresponde a uma Viatura do tipo Moto, possui os atributos herdados de Viatura.

3.6.2 Carro

Atributos

- ArrayList<Cliente> waitList;
Lista da fila de espera do Clientes.

Esta classe corresponde a uma Viatura do tipo Carro, para além dos atributos herdados de Viatura, esta classe permite estabelecer uma fila de espera.

3.6.3 Carrinha

Esta classe corresponde a uma Viatura do tipo Carrinha, possui os atributos herdados de Viatura.

3.7 Viagem

Atributos

- Localizacao inicioL
Localização do inicio da viagem.
- Localizacao fimL
Localização do final da viagem.

- int classificação
Classificação da viagem.
- double tempo
Duração da viagem.
- double preco
Preço da viagem.
- Calendar inicioT
Data do inicio da viagem.
- Calendar fimT
Data do fim da viagem.

Esta classe corresponde a uma Viagem. Esta classe só tem métodos mais comuns.

3.8 CatUtilizadores

Atributos

- TreeSet<Utilizador> catalog
TreeSet com utilizadores.

Esta classe corresponde a um catálogo de utilizadores.

3.9 CatViaturas

Atributos

- TreeSet<Viatura> catalog
TreeSet com viaturas.

Esta classe corresponde a um catálogo de viaturas.

4. *Perspectivas de melhorias e trabalho futuro*

Uma aplicação deste género é sempre um produto in acabo, necessidades mudam, por isso o desenvolvimento da mesma foi feito de maneira a que uma futura alteração seja o mais fácil possível, não necessitando de refazer toda a aplicação. Um exemplo prático seria o acrescentar de um novo tipo de veículo, bastando para tal acrescentar a sua classe e acrescentar a sua opção no menu do Motorista.

5. *Conclusão*

Ao longo da realização deste trabalho fomos confrontados com algumas dificuldades, tendo sido a falta de experiência neste novo paradigma/linguagem o maior dos obstáculos.

Mas, de uma forma geral as dificuldades foram superadas, culminando num trabalho minimamente completo do qual nos podemos orgulhar e do qual retiramos conhecimento não só para realização da UC em questão, mas também para a nossa vida enquanto futuros Engenheiros Informáticos.