

Ficha Prática #06

6.1 Objectivos

1. Praticar a modelação com **Diagramas de Sequência** e **Diagramas de Classe**.
2. Praticar a utilização de OCL.

6.2 Exercícios

Para os exercícios abaixo propostos analise os enunciados e responda às questões criando os respectivos diagramas.

6.2.1 Empresa de Transportes

Considere o diagrama de classes apresentado na Figura 6.1, que representa uma solução para uma empresa de transportes públicos e responda às seguintes questões:

1. Acrescente ao diagrama a seguinte restrição:

Não podem existir dois movimentos com o mesmo número numa instância de TabMovimentos.

2. Escreva um **Diagrama de Sequência** para o método da classe AutoInt:

```
String[*] comTantasParagens(int n)
```

que determina os códigos de todos os bilhetes que fizeram viagens (Movimentos) com um número de paragens igual ao valor *n* dado como parâmetro.

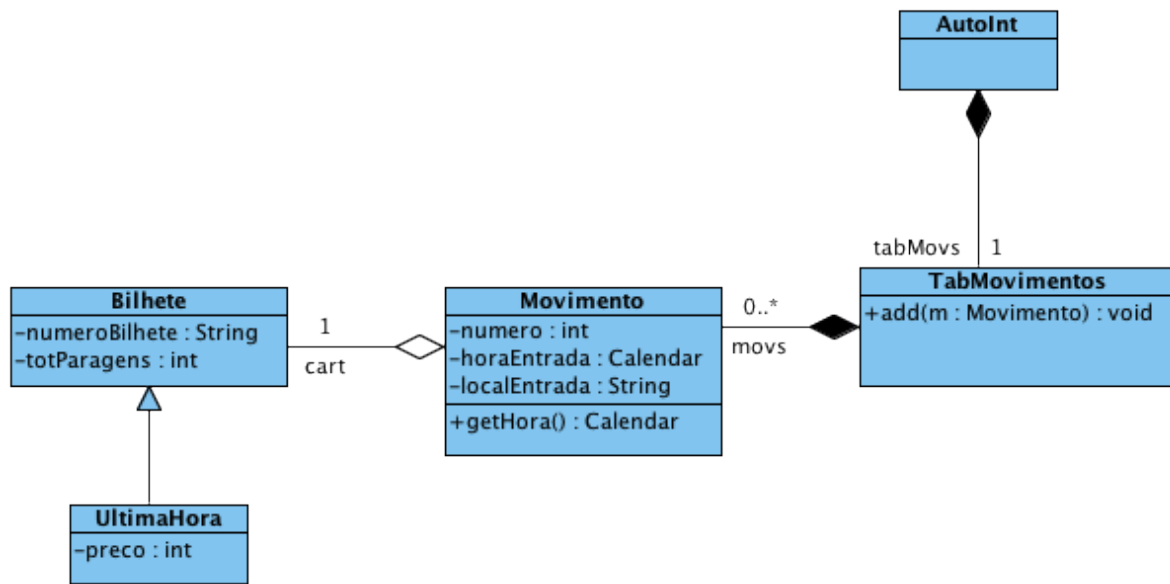


Figura 6.1: Diagrama de classes para o Exercício 6.2.1

3. Escreva um **Diagrama de Sequência** para o método da classe `AutoInt`:

```
Bilhete [*] inicioEm(String local)}
```

que determina todos os bilhetes que fizeram viagens com início em `local`.

6.2.2 Parques de Estacionamento

Considere que se pretende modelar um sistema de administração de parques de estacionamento. Neste sistema os clientes, para que possam estacionar nos diversos parques, devem possuir um identificador que está associado a uma determinada viatura. Caso um cliente tenha mais do que uma viatura é necessário que adquira tantos identificadores quantas as viaturas que possui. Sempre que um identificador é detectado num dado parque, é registado a hora e data de entrada e a hora e data de saída do parque. Cada parque tem uma tabela de preços baseada no tipo de viatura. Essa informação é fornecida pela associação que existe entre o identificador e a viatura.

A equipa de análise e o cliente desenvolveram, em conjunto, o modelo de domínio apresentado na Figura 6.2. Sabendo que durante a análise de requisitos ficou definido que deverá ser possível a um cliente pesquisar, e listar, os seus movimentos nos diversos parques, bem assim como obter os extractos de conta mensais para um determinado identificador. Tendo, a partir dos Use Case, sido possível identificar a

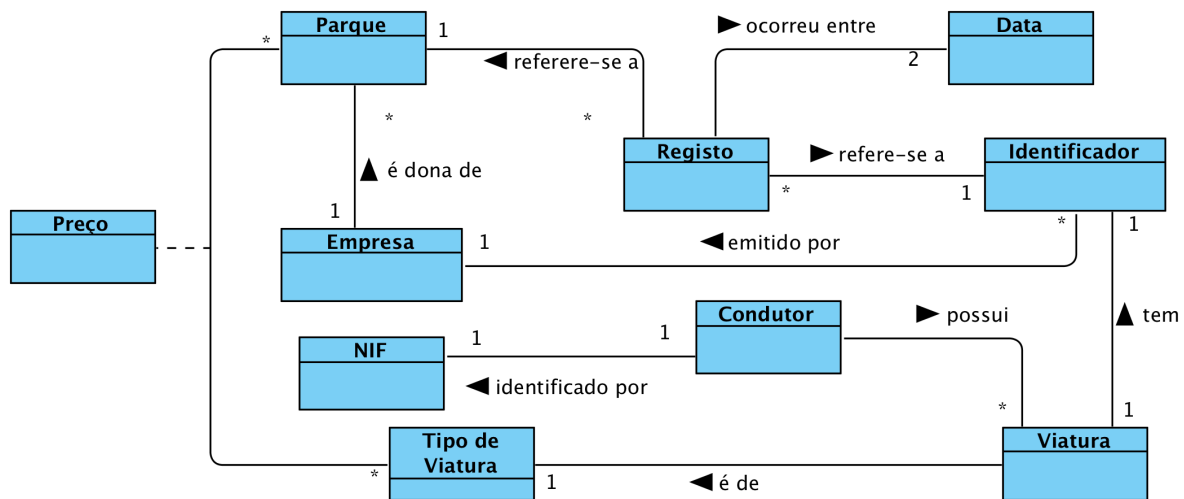


Figura 6.2: Modelo de Domínio para o Exercício 6.2.2

necessidade de implementar as operações referidas nas alíneas (3) a (6) abaixo e, finalmente, sabendo que:

- todas as associações derivadas do Modelo de Domínio apresentado na Figura 6.2 deverão ser agregações;
- a direcção a utilizar nas associações deverá ser definida em função dos métodos pedidos nas alíneas (3) e (4);
- para simplificar, numa primeira abordagem, a camada de negócio é implementada pela classe *GesPark* que possui (por composição) associações qualificadas para objectos *Viatura*, *Registo* e *Parque*.

Responda às seguintes questões:

1. Construa o **Diagrama de Classes** que permite capturar a informação fornecida no Modelo de Domínio. Seja o mais completo possível na construção do mesmo, identificando as relações, nomeando-as e colocando as multiplicidades respectivas.
2. Acrescente ao diagrama da alínea anterior, caso seja necessário, o **OCL** relativo à seguinte restrição:

Todos os registos de um parque são relativos a viaturas registadas no sistema.

3. Construa o **Diagrama de Sequência** para a operação da classe *GesPark* que, dado um identificador e um intervalo de datas, determina o montante a pagar pelo cliente nesse intervalo de datas.

A operação deverá ter como pré-condição a existência de uma viatura com o identificador dado. Acrescente-a ao Diagrama de classes.

4. Construa o **Diagrama de Sequência** para a operação da classe `GesPark` que, dado um identificador, determina o total de parques visitados pela viatura a que este está associado.

A operação deverá ter como pré-condição que o identificador existe e como pós-condição que o resultado é não negativo. Acrescente-as ao Diagrama de classes.

5. Construa o **Diagrama de Sequência** da operação que permitirá calcular, para todos os identificadores de um cliente identificado pelo seu NIF, qual o montante global a pagar num determinado intervalo de datas.

A operação deverá ter como pré-condição que o cliente existe no sistema. Acrescente-a ao Diagrama de classes.

6. Construa o **Diagrama de Sequência** para a operação da classe `GesPark` que deve verificar se não existem duas viaturas com o mesmo identificador.

A operação deverá ter como pós-condição que o resultado será verdadeiro se não existirem duas viaturas com o mesmo identificador. Acrescente-a ao Diagrama de classes.