

Programação Orientada a Objetos

Prof. Márcio Miguel Gomes



JESUÍTAS BRASIL



UNISINOS

Nomenclatura

- Ao definir um identificador qualquer (nome de variável, função, método, estrutura, atributo, etc) deve-se sempre utilizar um nome que seja **autoexplicativo**.
- Evite usar identificadores **genéricos** como a, x, y, aux, flag.
- Variáveis i, j devem ser utilizados em contextos restritos, como variáveis de controle para estruturas de repetição

Nomenclatura

- Linguagens como C, C++, C# e Java utilizam a notação **camelCase**, que **não** permite **espaços** entre palavras, inicia com a **primeira palavra** toda em letra **minúscula** e a cada **nova palavra**, usa apenas a **primeira letra maiúscula**.
- Ex: nome, cpf, dataNascimento, nomeDoPai, nomeDaMae, enderecoCobranca, imprimeNotaFiscal(), consultaSaldo()

Nomenclatura

- A linguagem Python utiliza a notação **snake_case**, que **não** permite **espaços** entre palavras, **todas** as palavras são em letra **minúscula** e a cada **nova palavra** é utilizado o separador **underscore** “_”.
- Ex: nome, idade, nome_da_mae, salario_por_hora, valor_maximo

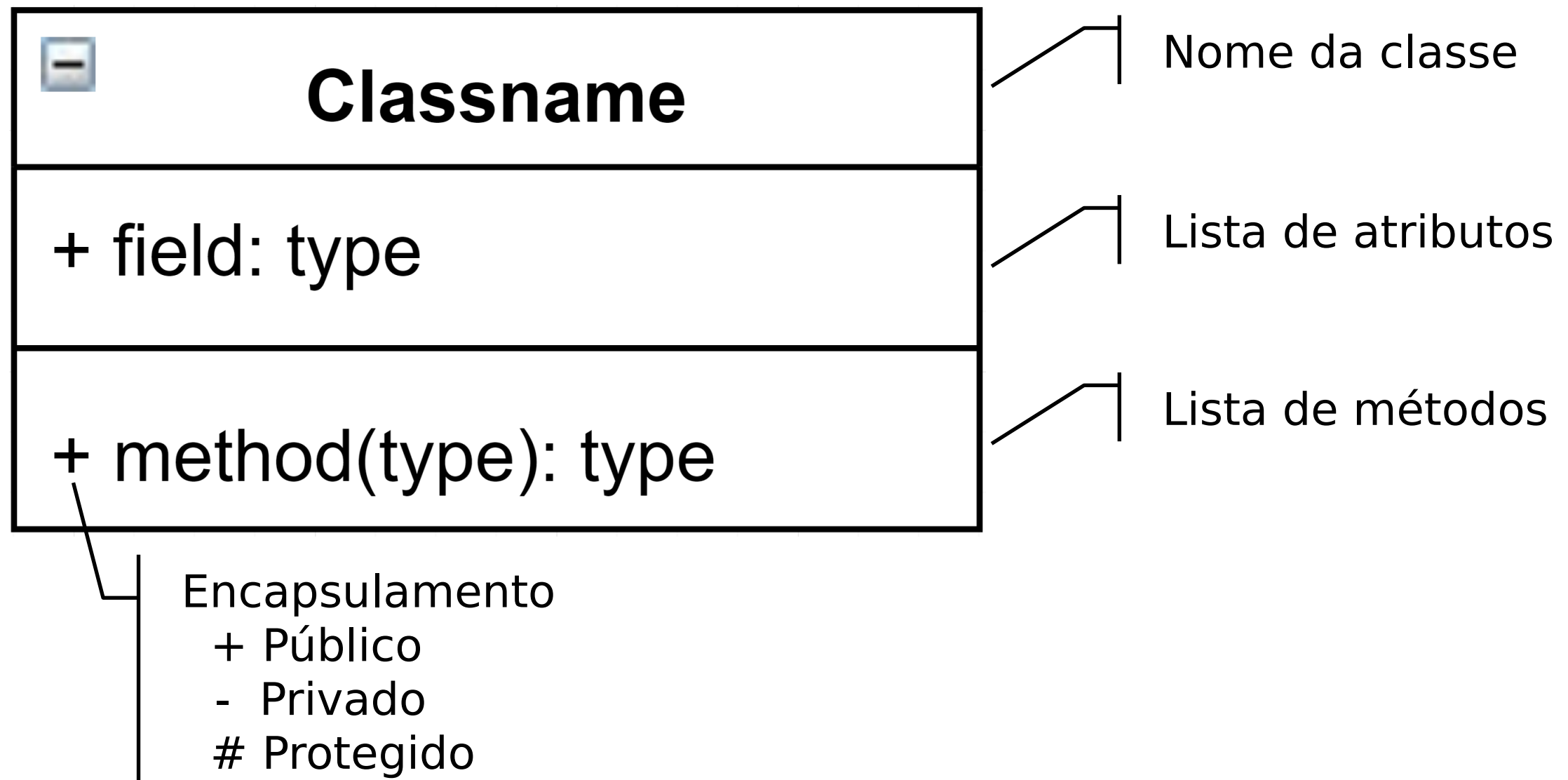
Nomenclatura

- Para nomear **classes**, utilize a notação **PascalCase**, que **não** permite **espaços** entre palavras, a **primeira letra** de cada palavra deve ser **maiúscula** e as **demais letras** devem ser **minúsculas**.
- As classes **primárias** devem ser escritas no **singular**.
- Ex: Pessoa, Endereco, TipoEndereco, NotaFiscal, Item, Produto
- O **relacionamento** entre classes deve conter os **nomes das classes** envolvidas. Normalmente, a **primeira** classe fica no **plural** e a **segunda** no **singular**.
- Ex: EnderecosPessoa, ItensNotaFiscal

UML

- *Unified Modeling Language* é uma **padronização** internacional para **representar** diversos **diagramas** em um projeto de software **orientado a objetos**.
- O diagrama mais utilizado é o **diagrama de classes**, que documenta a estrutura de uma classe definindo seu **nome**, **atributos**, **métodos** e **relacionamentos** com outras classes.

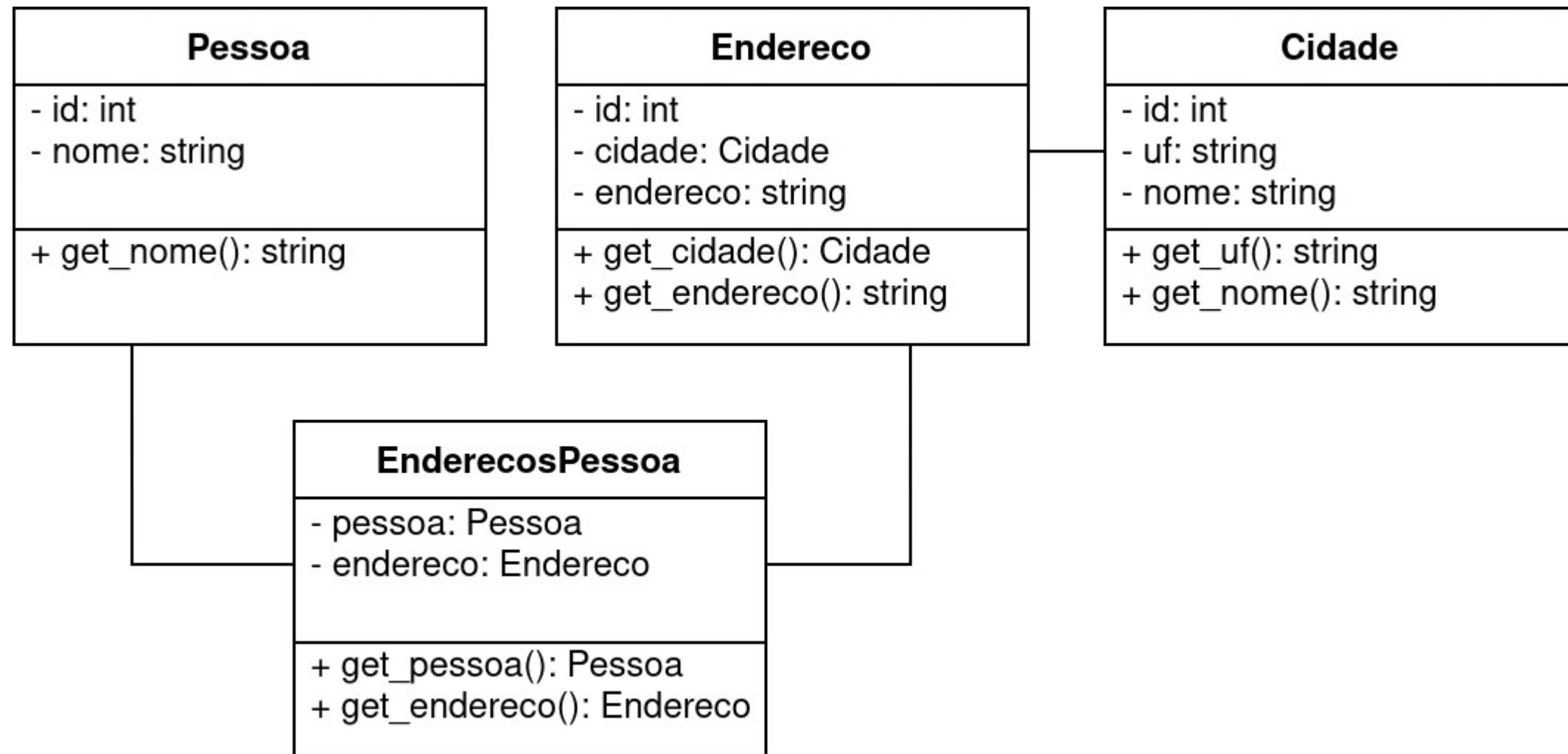
Diagrama de Classes



Relacionamento

- Uma **classe** pode se **relacionar** com outras de **diversas maneiras**.
- Na UML, a **associação** entre classes é representada por **linhas** e **setas específicas**.
- Por questão de **simplicidade**, utilizaremos apenas uma **linha contínua** para ligar as classes que possuem algum **relacionamento**.

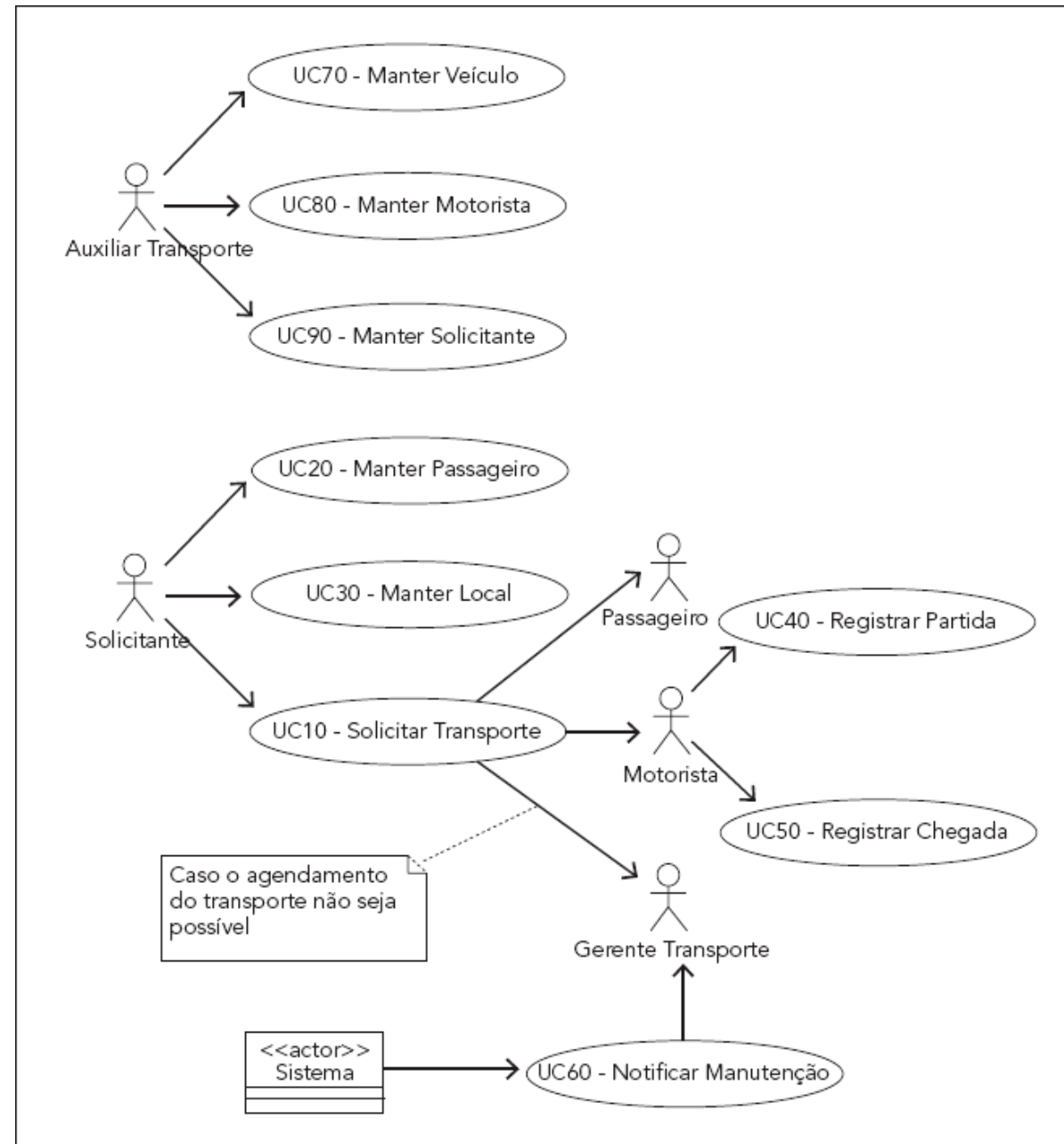
Relacionamento



Caso de Uso

- Representa a **sequência** de **interação** entre dois ou mais **atores** através de **troca de informações, verificações, processamentos e ações** realizadas para **execução** de uma **tarefa** específica.
- Pode ser representado de forma **textual** como um **algoritmo** simplificado, ou de forma **gráfica** através do **diagrama** de caso de uso da UML.

Caso de Uso - Diagrama



Caso de Uso - Textual

- Empréstimo de livros em uma biblioteca
 - O **leitor** chega ao balcão de atendimento da biblioteca e diz ao **atendente** que deseja **retirar** um ou mais **livros** da biblioteca.
 - O **atendente solicita** ao **leitor** sua carteira de **identificação**
 - O **leitor fornece** sua carteira de **identificação**.
 - O **atendente informa** ao sistema a **identificação** do leitor.
 - O sistema **exibe** o **nome** do **leitor** e sua **situação**.
 - O **atendente solicita** os **livros** a serem emprestados.
 - O **leitor entrega** os **livros** para o **atendente**.
 - O **atendente informa** ao sistema o código de **identificação** dos **livros**.
 - O sistema **informa** a **data de devolução** de cada **livro**.
 - O **atendente desbloqueia** os **livros** para que possam sair da biblioteca.
 - O **leitor sai** com os **livros**.

Atividade

- Faça um diagrama de classes UML que represente um sistema de gerenciamento de um hotel.
- As classes a serem detalhadas são Hotel, Quarto, Cliente e Reserva, além dos relacionamentos QuartosHotel e ReservasCliente.
- Defina os atributos, métodos e relacionamentos de cada classe conforme necessidade. Lembre-se de utilizar os princípios da orientação a objetos estudados na aula anterior, principalmente abstração e encapsulamento.
- Descreva textualmente um caso de uso para realizar uma reserva e outro caso de uso para realizar o *checkout*.