

# Modelagem de Software

Prof. Esp. Erinaldo Sanches Nascimento  
erinaldonascimento@yahoo.com.br

- Modelos de contexto
  - Perspectiva externa do software
- Modelos de interação
  - Perspectiva do diálogo entre o sistema e o usuário
- Modelos estruturais
  - Componentes, banco de dados, relacionamentos, diagrama de classes.
- Modelos comportamentais
  - Procedimentos dinâmicos do software

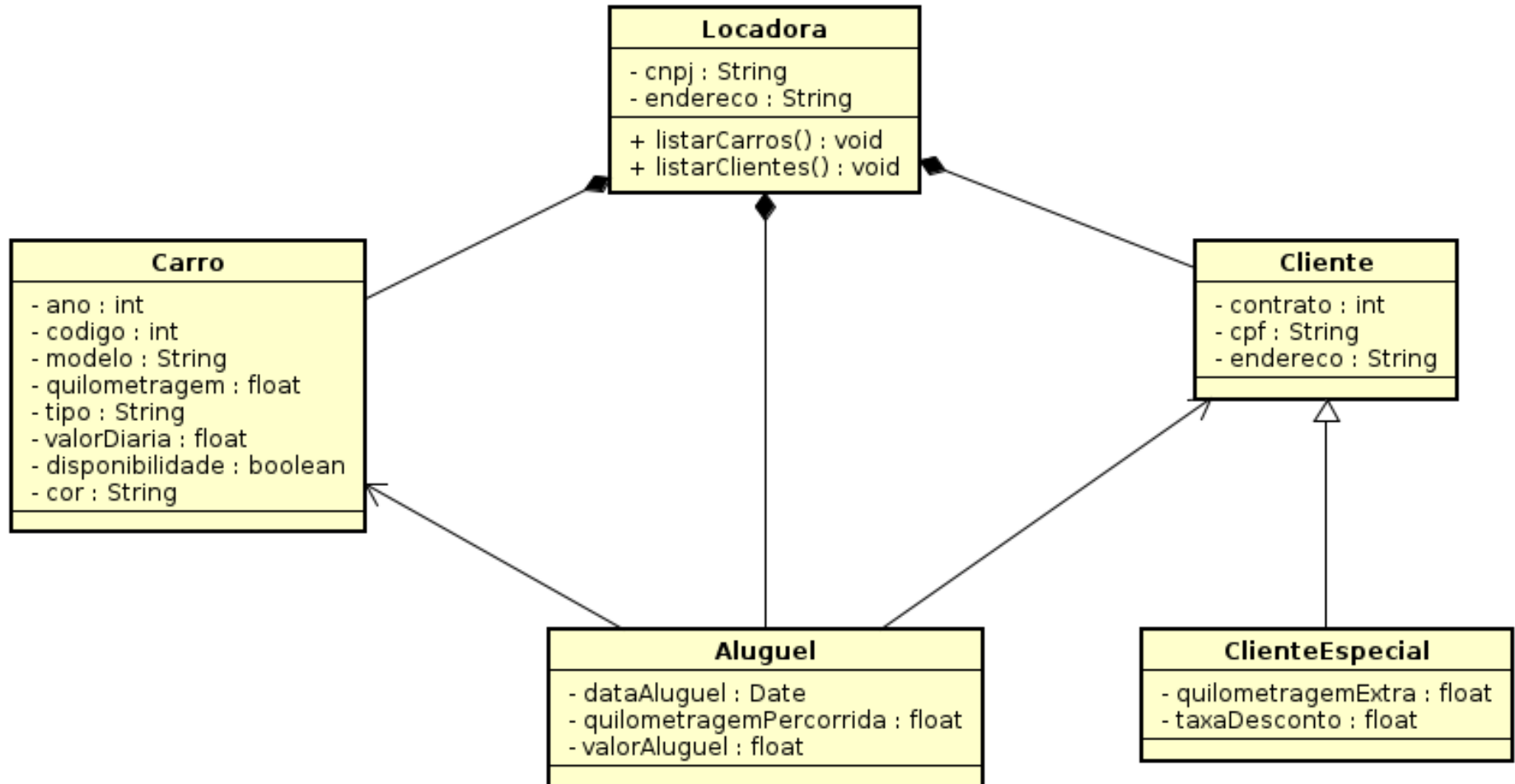
- Representam a organização, a disposição e ordem dos elementos essenciais que compõem o sistema.
- **Estático:** estrutura do projeto do sistema.
- **Dinâmico:** organização do sistema quando está em execução.

- Objetos – Classes
- Classes: conjunto de objetos com as mesmas características.



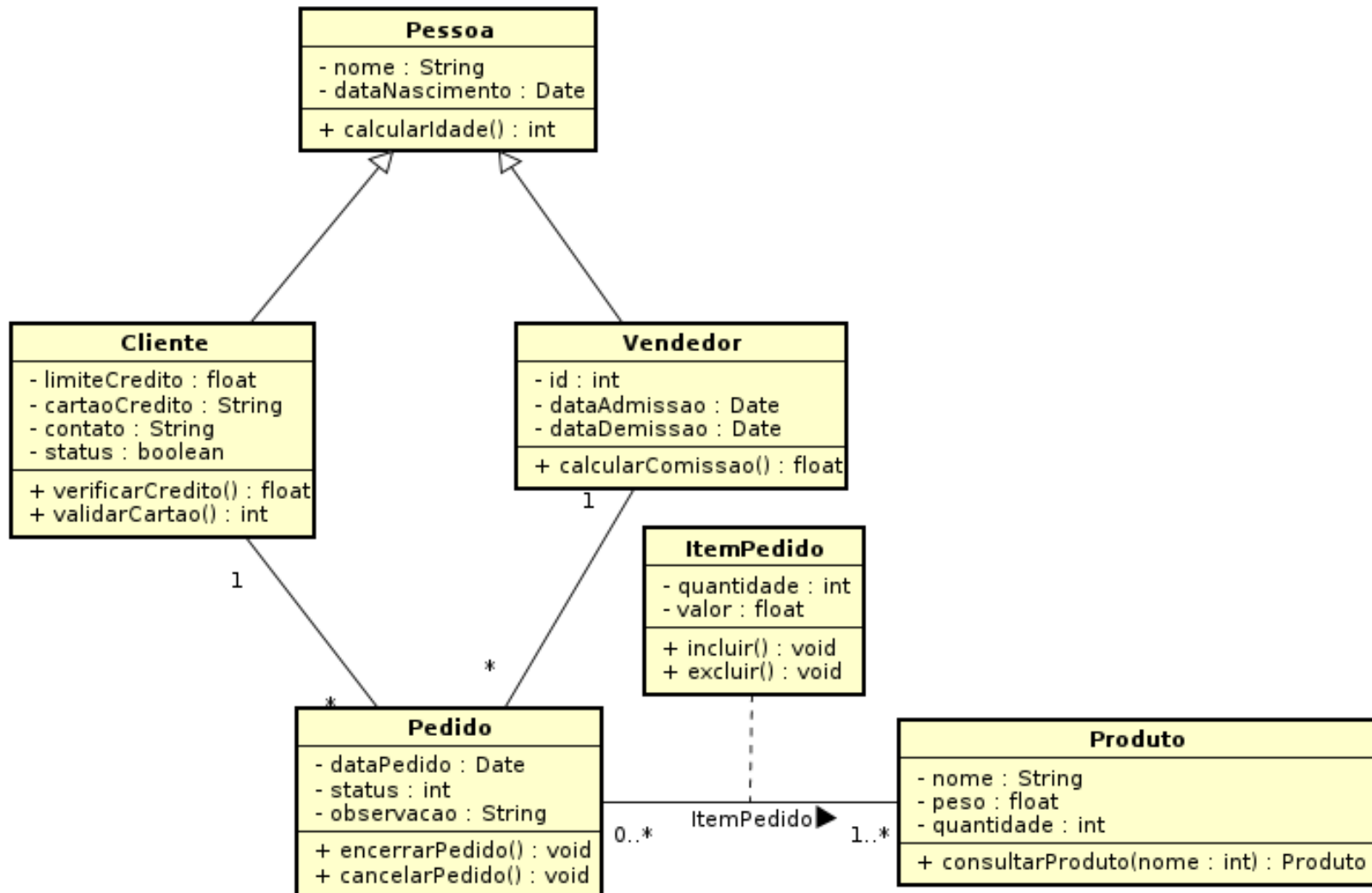
- Identificar os objetos.
- Classificá-los em classes específicas (considerar características semelhantes).

# Diagrama de Classes



- Associação (binária / unária / n-ária)
- Multiplicidade determina qual das classes envolvidas em uma associação fornece informações para as outras.

# Diagrama de Classes



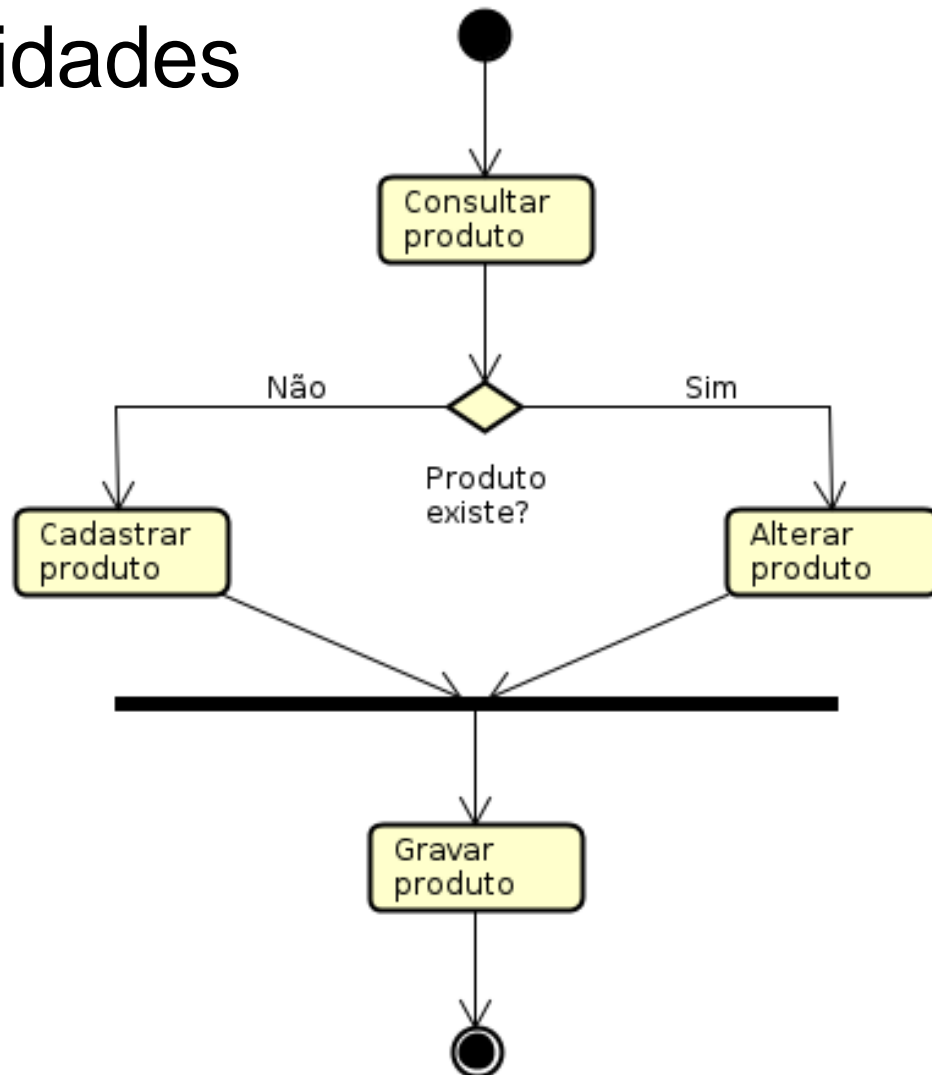


- Representam o comportamento dinâmico do sistema.
- Representa a perspectiva do comportamento dos dados mediante a ação das funções.
- Retrata os estados e os eventos que transformam esses estados.

- Modelagem orientada a dados:
  - DFD – Diagrama de Fluxo de Dados
  - DD – Dicionário de Dados
  - Diagrama de Atividades
- Modelagem orientada a eventos:
  - Diagrama de Estados

- Na programação orientada a objetos a execução do programa é direcionada por eventos.

## Diagrama de Atividades



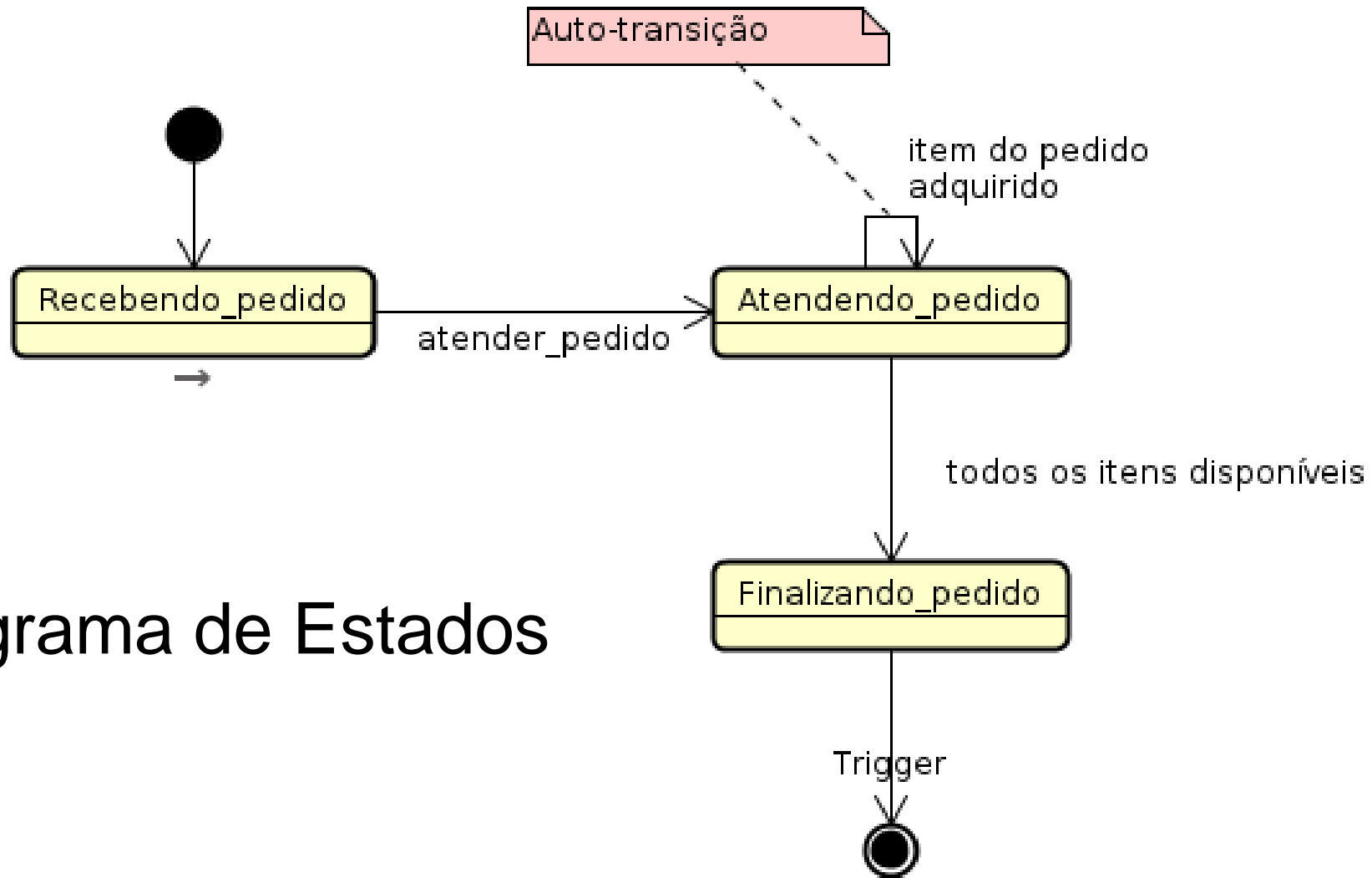


Diagrama de Estados

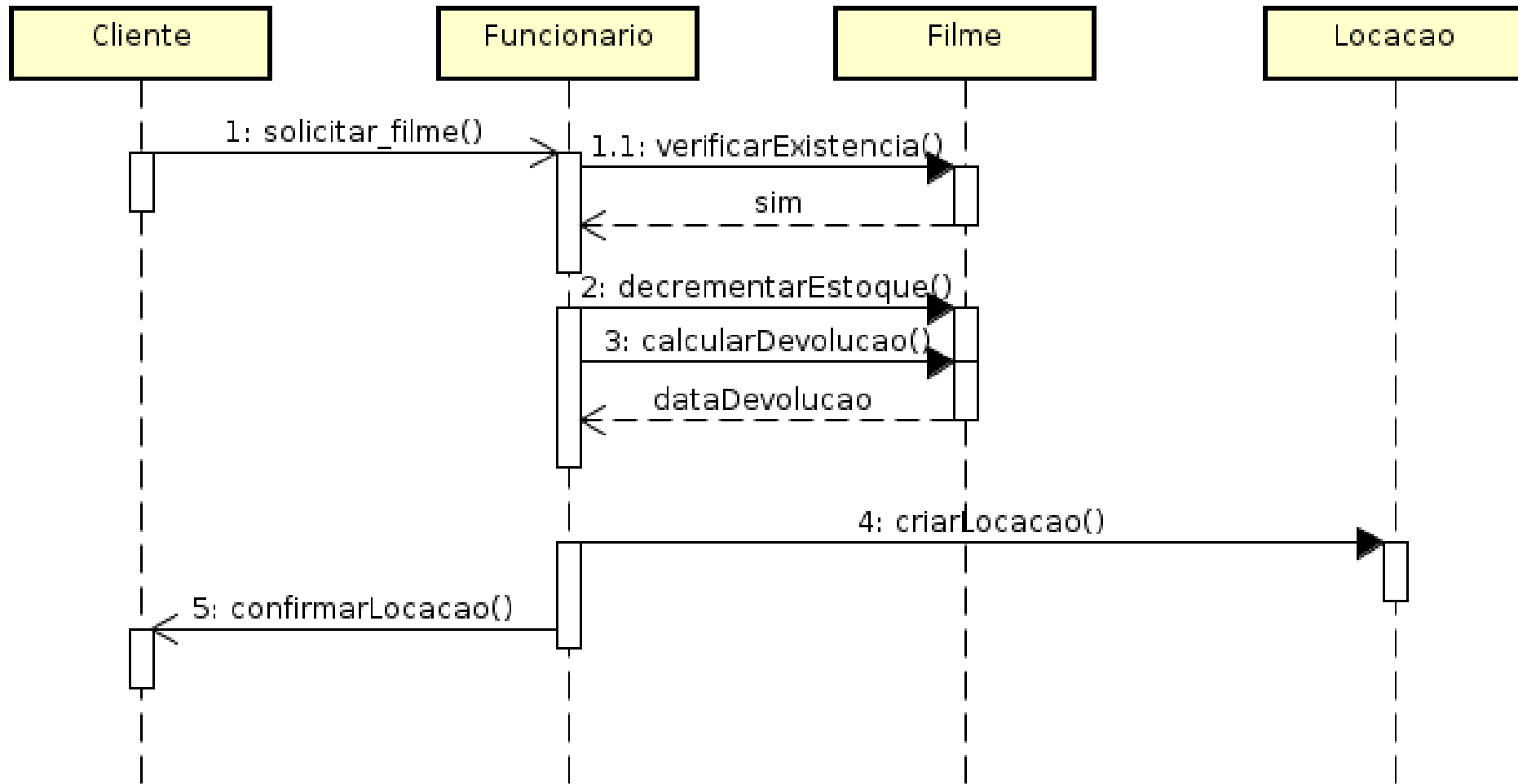


Diagrama de Sequência

- Modelos estruturais representam a organização e arquitetura do software;
- Diagramas de classe são utilizados para representar a estrutura estática das classes e seus relacionamentos;
- Os diagramas de estado são utilizados para modelar o comportamento de um software em resposta a eventos;
- O diagrama de sequência da UML pode modelar o fluxo de dados de um sistema.