

Engenharia de Software

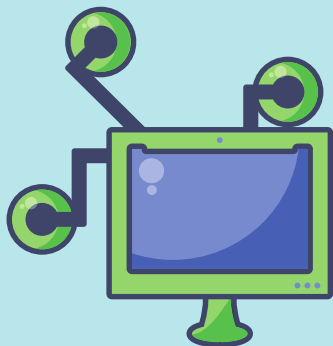


A UML não é uma linguagem de programação, mas uma linguagem de modelagem que tem, como meta, auxiliar os engenheiros de software a definirem as características do software, tais como seus requisitos, seu comportamento, sua estrutura lógica, a dinâmica de seus processos e até mesmo suas necessidades físicas em relação ao equipamento em que o sistema deverá ser implantado.

Modelagem de Sistemas

A modelagem de sistema é o processo de elaboração de modelos abstratos de um sistema.

Da mesma forma que os arquitetos elaboram plantas e projetos para que haja a construção de um edifício, os engenheiros de software criam os diagramas UML para auxiliarem os desenvolvedores de software a construírem o software.



Diagramas UML

A UML define, em sua versão 2.0, treze tipos de diagramas para uso na modelagem de software:

- 1. Diagrama de Casos de Uso;*
- 2. Diagrama de Classes;*
- 3. Diagrama de Objetos;*
- 4. Diagrama de Pacotes;*
- 5. Diagrama de Sequência;*
- 6. Diagrama de Comunicação;*
- 7. Diagrama de Máquina de Estados*
- 8. Diagrama de Atividade;*
- 9. Diagrama de Visão Geral de Interação;*
- 10. Diagrama de Componentes;*
- 11. Diagrama de Implantação;*
- 12. Diagrama de Estrutura Composta;*
- 13. Diagrama de Tempo ou de Temporização*

Diagramas mais utilizados

1

Casos de Uso

O diagrama de casos de uso é, dentre todos os diagramas da UML, o mais abstrato, flexível e informal. Guedes (2007)

2

Classes

Diagrama de Classe permite a visualização das classes que comporão o sistema com seus respectivos atributos e métodos e demonstra como as classes do diagrama se relacionam, complementam e transmitem informações entre si. Guedes (2009)

4

Sequência

É um diagrama comportamental que procura determinar a sequência de eventos que ocorrem em um determinado processo, identificando quais mensagens devem ser disparadas entre os elementos envolvidos e em que ordem. Guedes (2009)

3

Máquina de Estados

demonstra o comportamento de um elemento por meio de um conjunto finito de transições de estado, ou seja, uma máquina de estados. Guedes (2009)

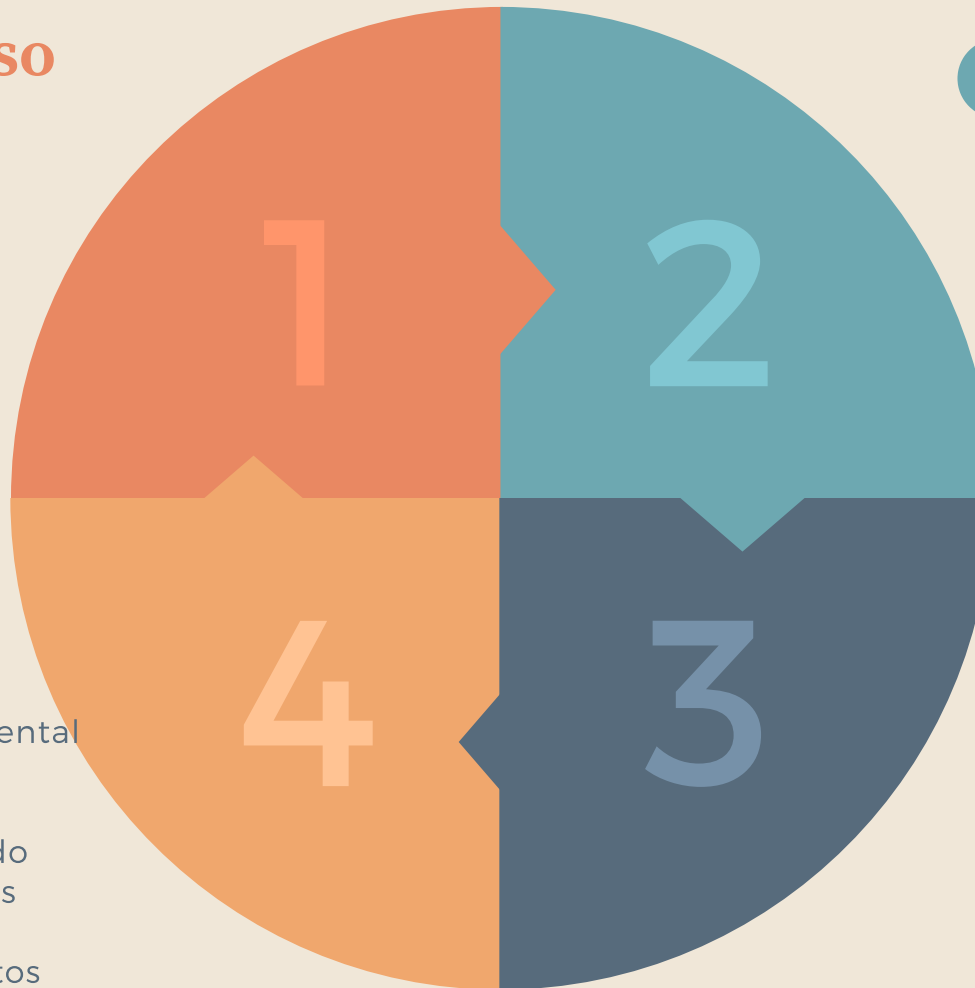


Diagrama de Casos de Uso

O principal objetivo desse diagrama é o de modelar as funcionalidades e serviços oferecidos pelo sistema, buscando, por meio de uma linguagem simples, demonstrar o comportamento externo do sistema a partir da perspectiva do usuário.

Este diagrama é composto por alguns elementos:

- Ator;
- Caso de Uso;
- Generalização/Especialização;
- Inclusão (<<include>>);
- Extensão (<<extend>>);

Diagrama de Casos de Uso

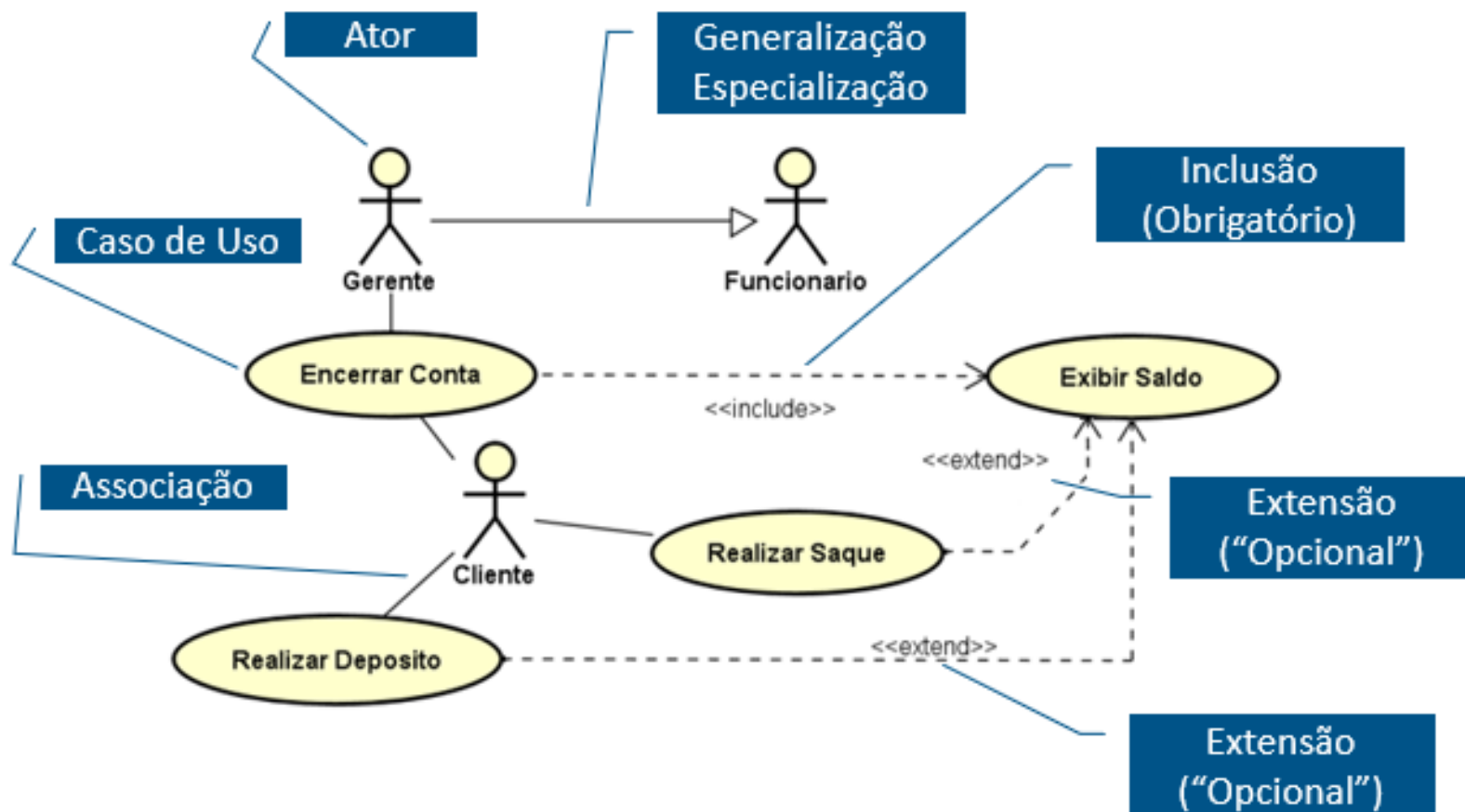


Diagrama de Classes

O Diagrama de classes possibilita a elaboração de grande parte dos diagramas de UML, uma vez que alicerça o sistema. Este diagrama, basicamente, é composto por classes e pelas associações entre as mesmas – relacionamentos entre as classes) (GUEDES, 2009).

Este diagrama é composto por alguns elementos:

- Classes;
- Relacionamentos:
- Associação;
- Agregação;
- Composição;
- Generalização/Especialização;
- Classe Associativa;

Diagrama de Classes

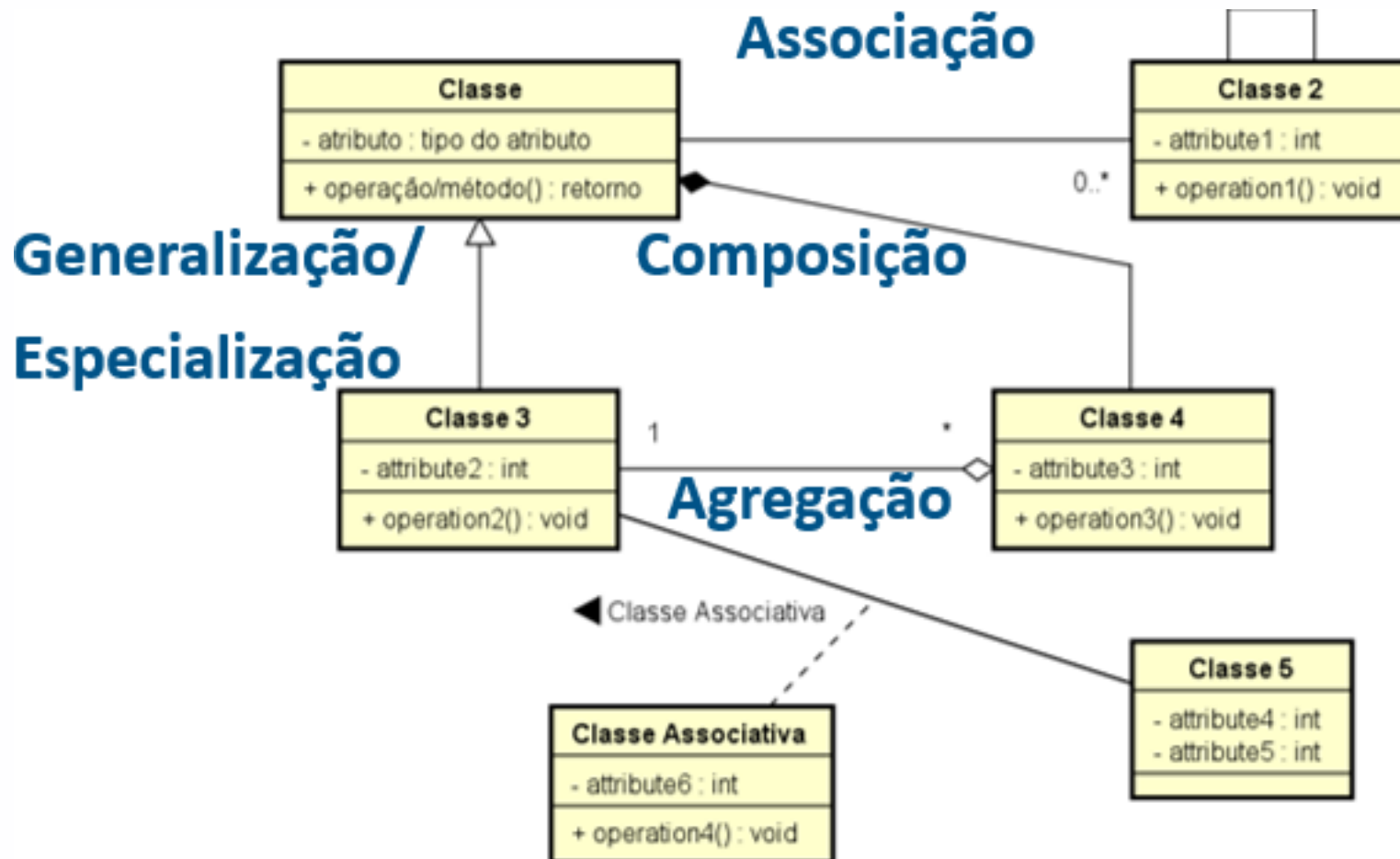


Diagrama de Sequência

“O diagrama de sequência procura determinar a sequência de eventos que ocorrem em um determinado processo, identificando quais mensagens devem ser disparadas entre os elementos envolvidos e em que ordem” (GUEDES, 2011, p. 192).

Este diagrama é composto por alguns elementos:

- Ator;
- Linha de vida (lifelines);
- Foco de Controle/Ativação;
- Mensagens;

Diagrama de Sequência

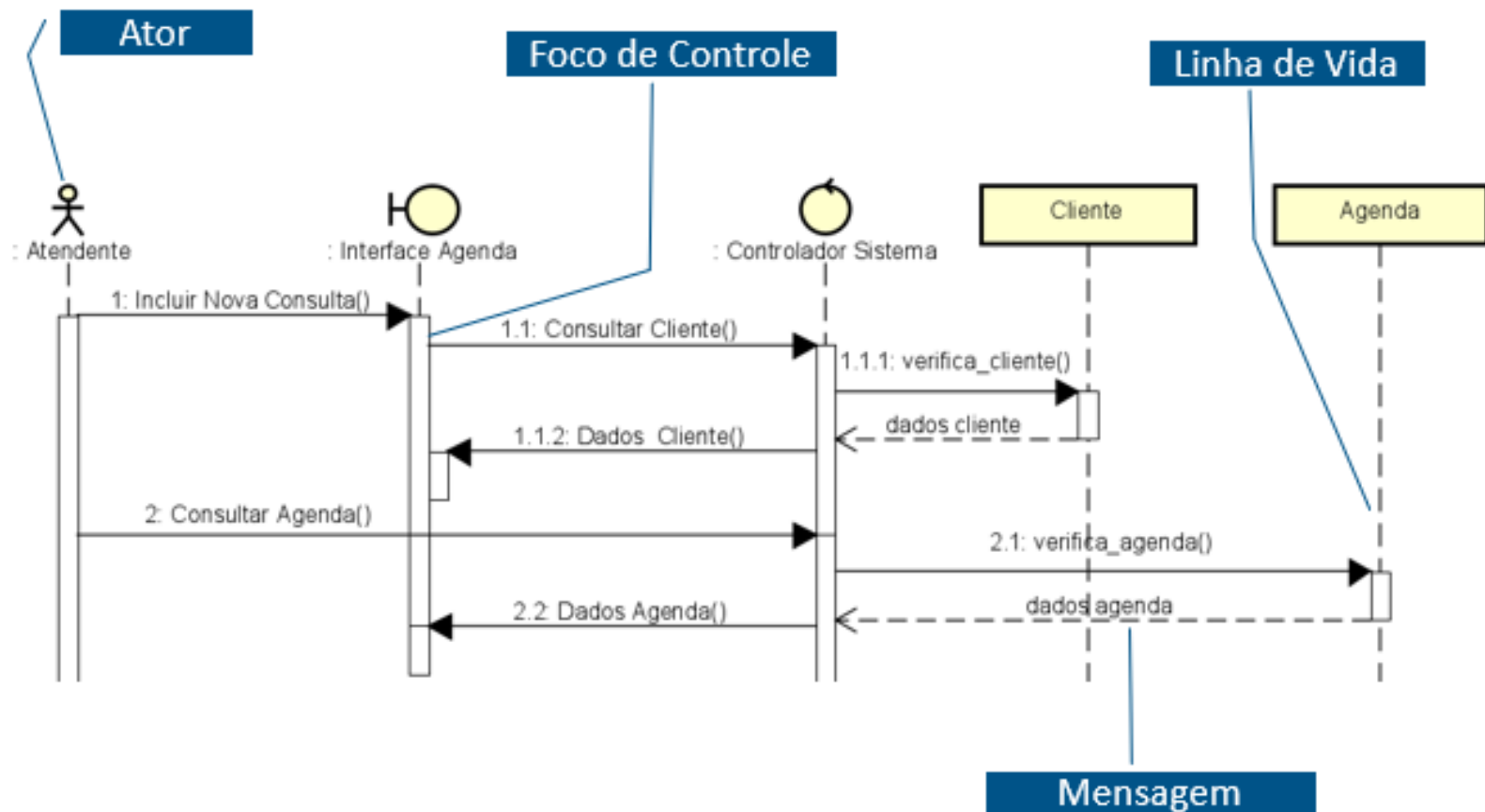


Diagrama de Máquina de Estado

O diagrama de máquina de estados é um diagrama de comportamentos. Ele pode ser usado para especificar o comportamento de vários elementos, seja uma instância de uma classe ou um diagrama de caso de uso, por exemplo. (GUEDES, 2011).

Este diagrama é composto por alguns elementos:

- Estado Inicial;
- Estado;
- Transições;
- Estado Final;

Diagrama de Máquina de Estado

