

Análise orientado a objetos

Fundamentos da UML

Ma. Vanessa Matias Leite

- Unidade de Ensino: 01
- Competência da Unidade: Compreender os fundamentos da UML
- Resumo: Conhecer os diagramas que compõem a UML e as fases do processo unificado.
- Palavras-chave: UML; processo unificado; paradigma orientado a objetos;
- Título da Teleaula: Fundamentos da UML
- Teleaula nº: 01

Características da UML

Paradigma Orientado a Objetos

- Linguagens como JAVA, C++, C#, PHP;
- Relacionamento entre classes e objetos e o relacionamento entre eles:
 - Herança;
 - Polimorfismo,
 - Agregação
 - composição

Paradigma Orientado a Objetos



Fonte: <https://pixabay.com/pt/>

Classe

Define o comportamento de seus objetos - através de métodos - e os estados possíveis destes objetos - através de atributos.

Objeto

Instância de um classe.

Herança

As classes compartilham seus atributos, métodos e outros membros da classe entre si.

Polimorfismo

Métodos que têm a mesma assinatura, mas comportamentos distintos.

Encapsulamento

Proibição do acesso ao direto ao estado do objeto.

Características da UML

- Combina os conceitos comuns de linguagens OO;
- Compatível com o desenvolvimento de software desde os requisitos até as etapas finais do desenvolvimento;
- Compatível com diversos escopos;

Objetivos da UML

- Modelar diferentes linguagens e situações;
- Padrão para o desenvolvimento de software;
- Simplicidade;

Modelos

- Capturar e definir com precisão os requisitos do software;
- Auxiliar o início do projeto do sistema;
- Solução que contenha as decisões de projeto;
- Explorar diferentes soluções;
- Permitir o fácil entendimento de projetos complexos.

Fluxo de desenvolvimento

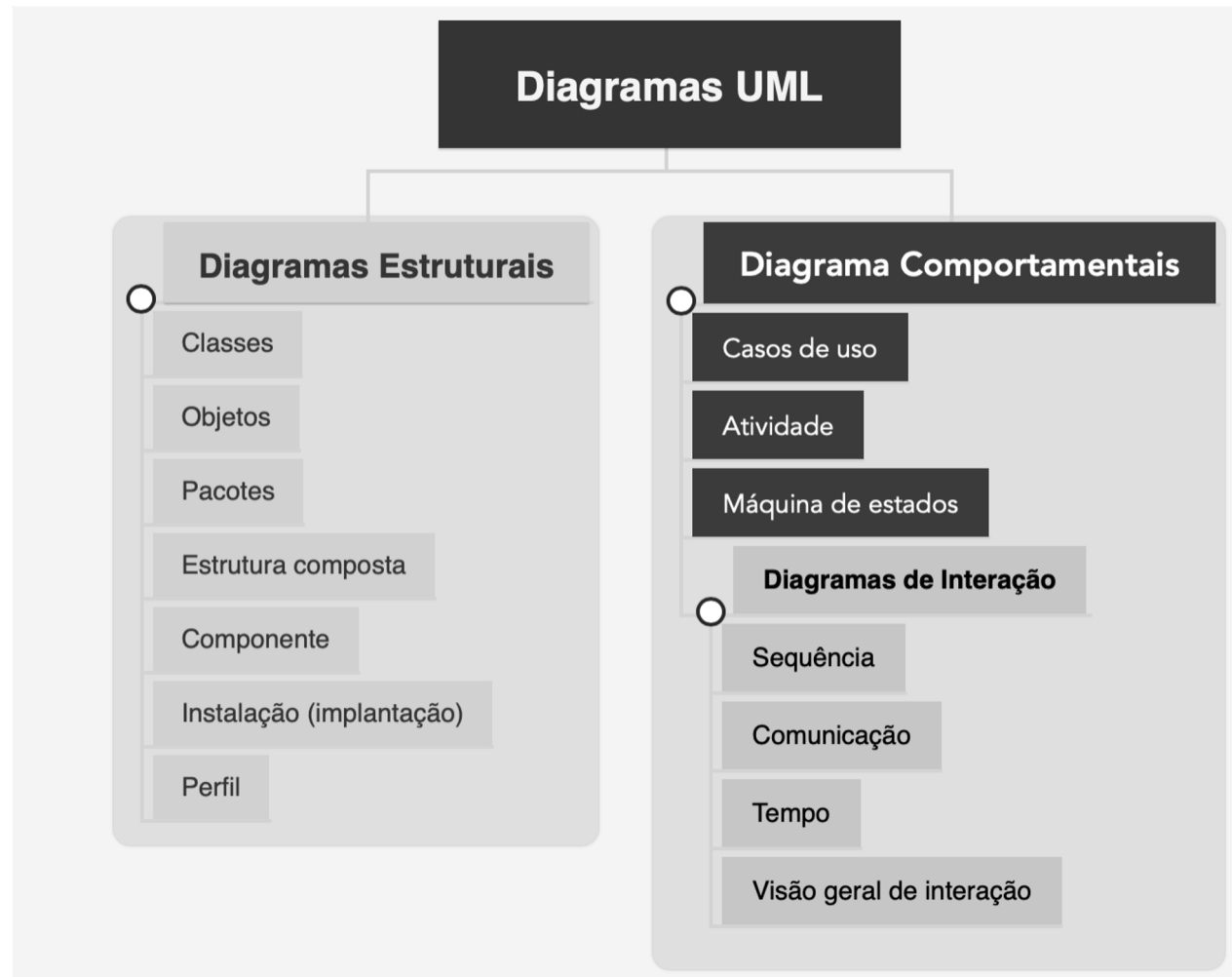
- Pode ser utilizada em qualquer fluxo de desenvolvimento;
- Apresentação visual da semântica do sistema;
- Contexto;

Diagramas UML

Nível de abstração

| Nível de Abstração | Objetivo do Diagrama |
|--------------------|--|
| ALTO | Ser claro e simples, apresentar os conceitos ao cliente para tomada de decisão |
| MÉDIO | Guiar o desenvolvimento apresentado, sem detalhar demais, as classes, os objetos e as interações |
| BAIXO | Demonstrar como deve ser desenvolvido o sistema propriamente dito. Necessita de diagramas e modelos com a especificação completa de cada módulo, interação e outras informações que possam ser necessárias |

Fonte: Livro texto

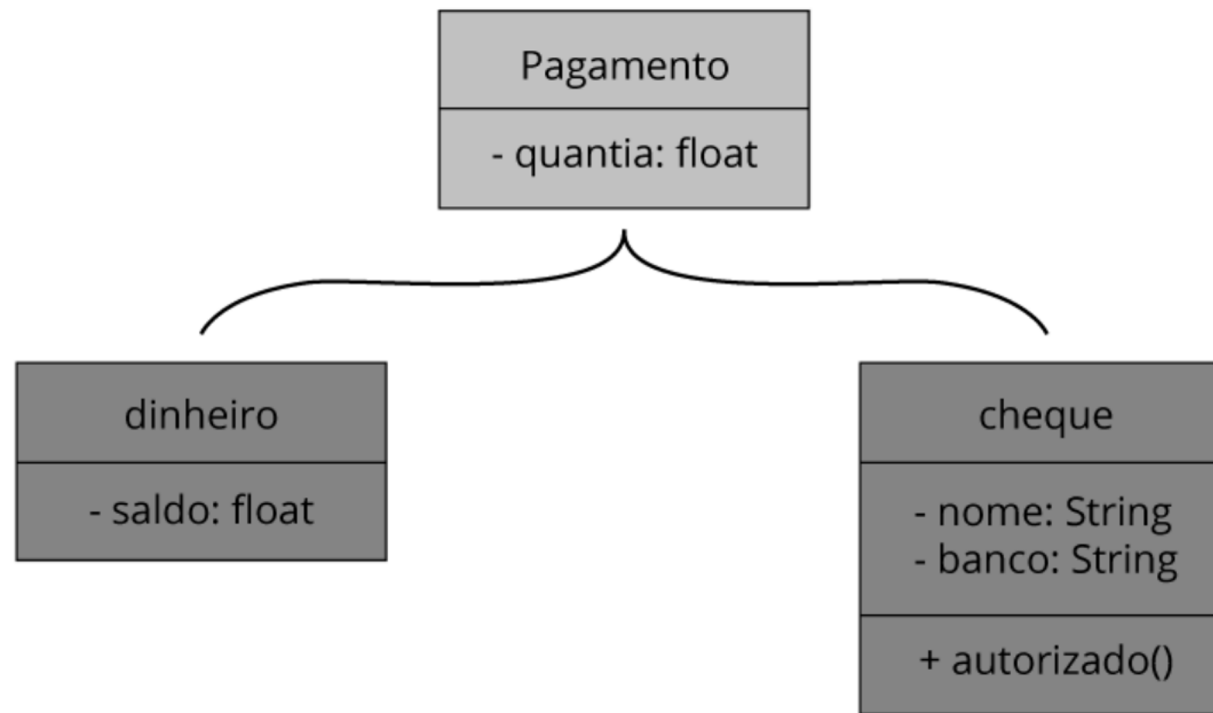


Fonte: Livro texto

Diagrama de classes

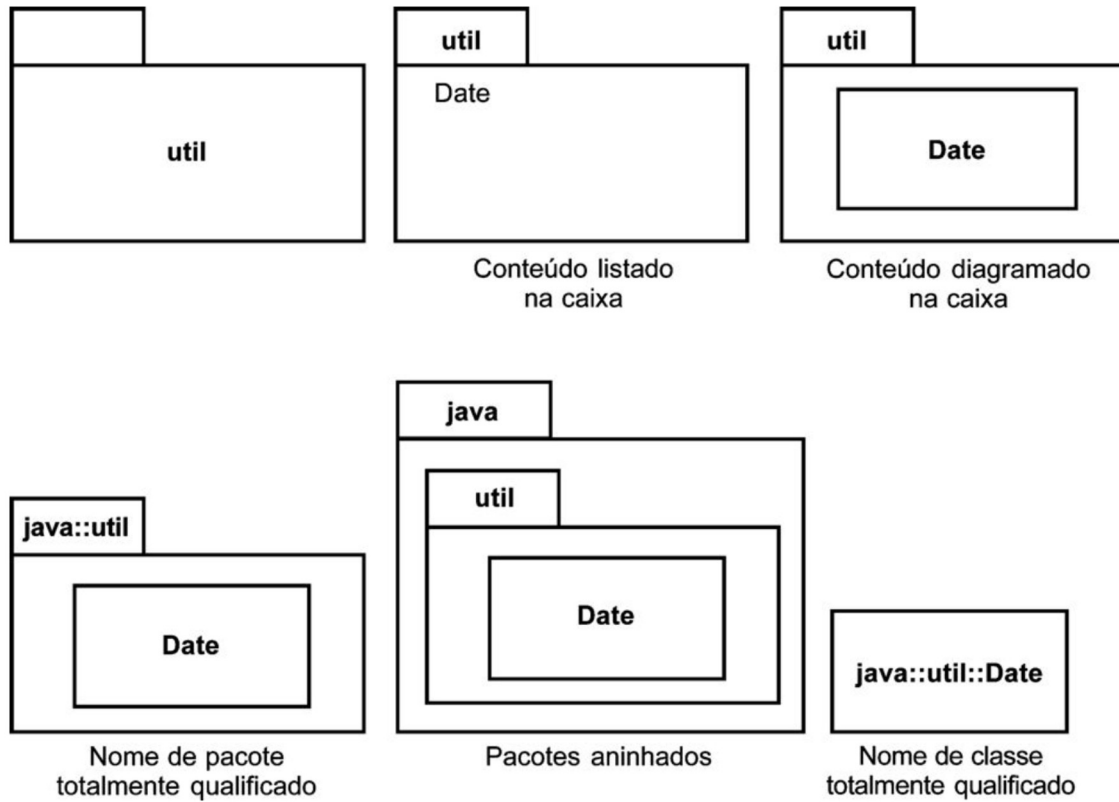
- Classe:
 - Atributos;
 - Métodos;
- Relacionamento:
 - Associação;
 - Herança;
 - Polimorfismo;

Diagrama de classes



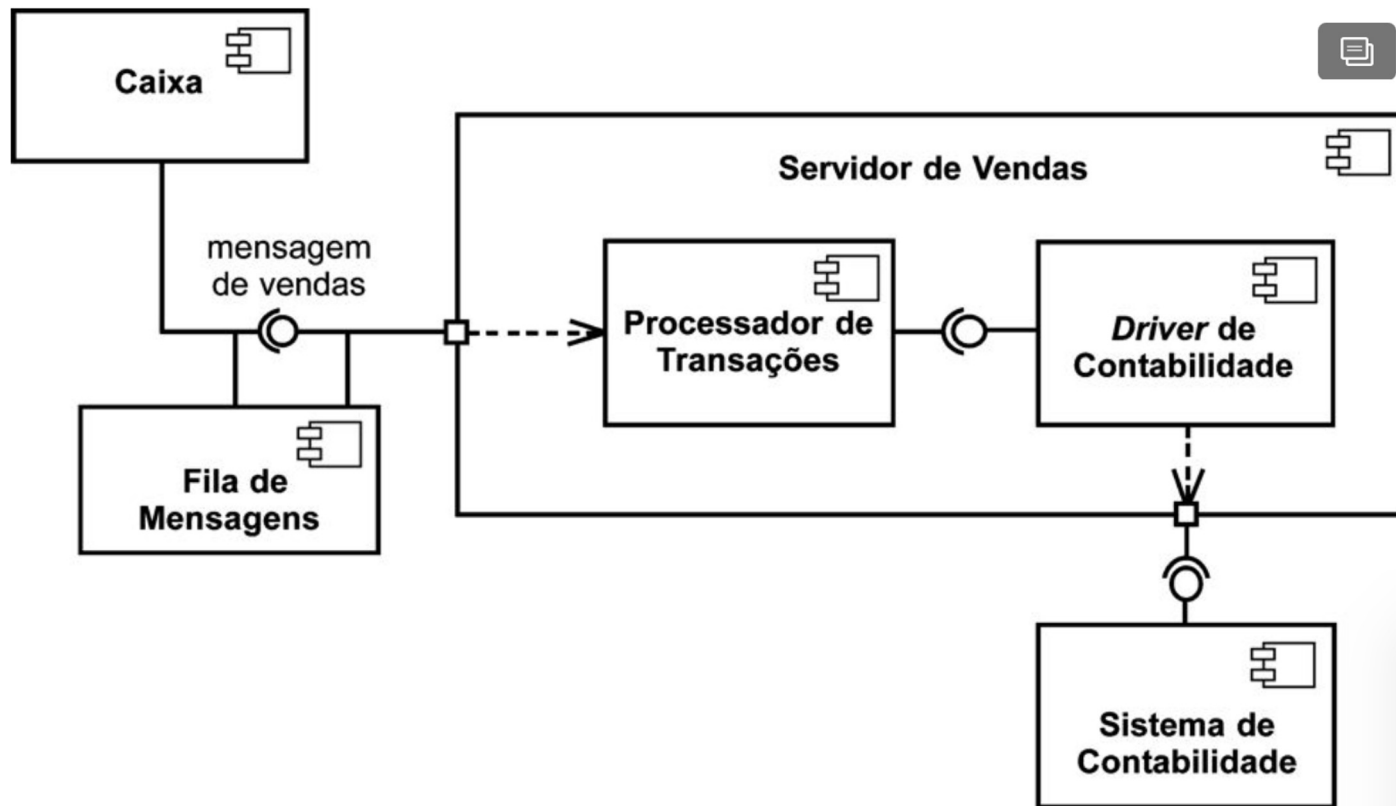
Fonte: Livro texto

Diagrama de Pacotes



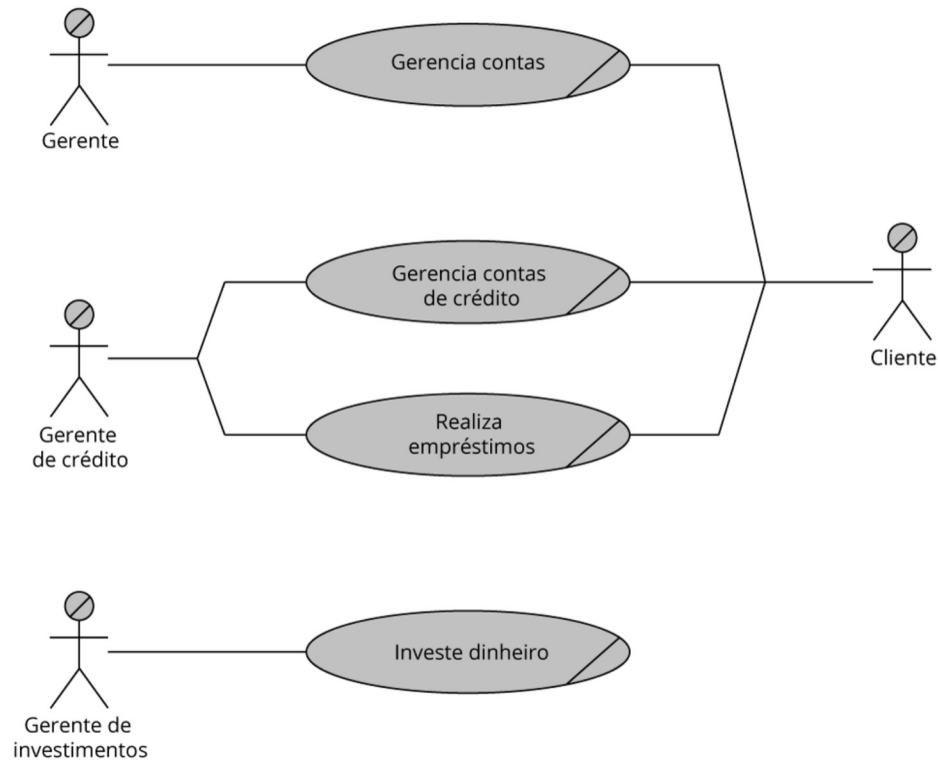
Fonte: Fowler (2005)

Diagrama de Componentes



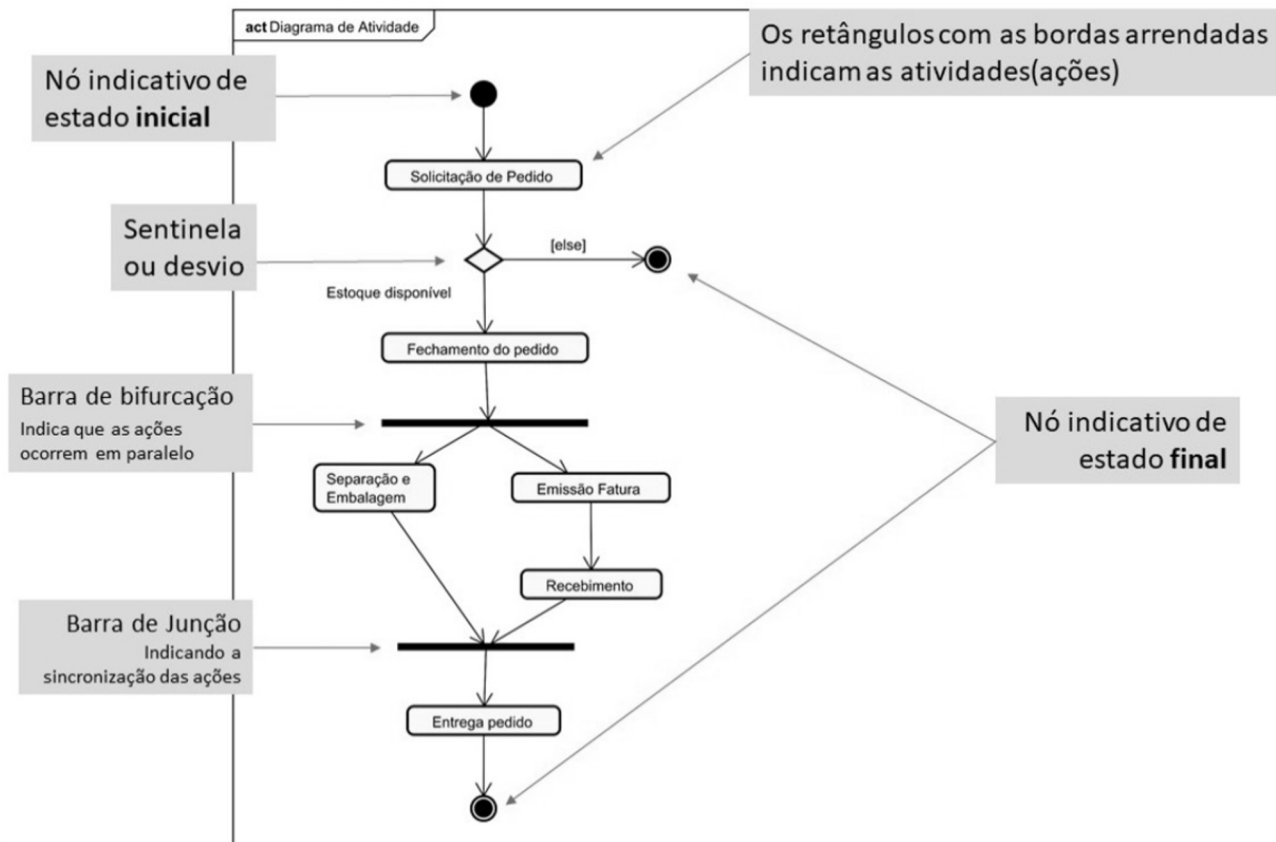
Fonte: Fowler (2005)

Diagrama de caso de uso

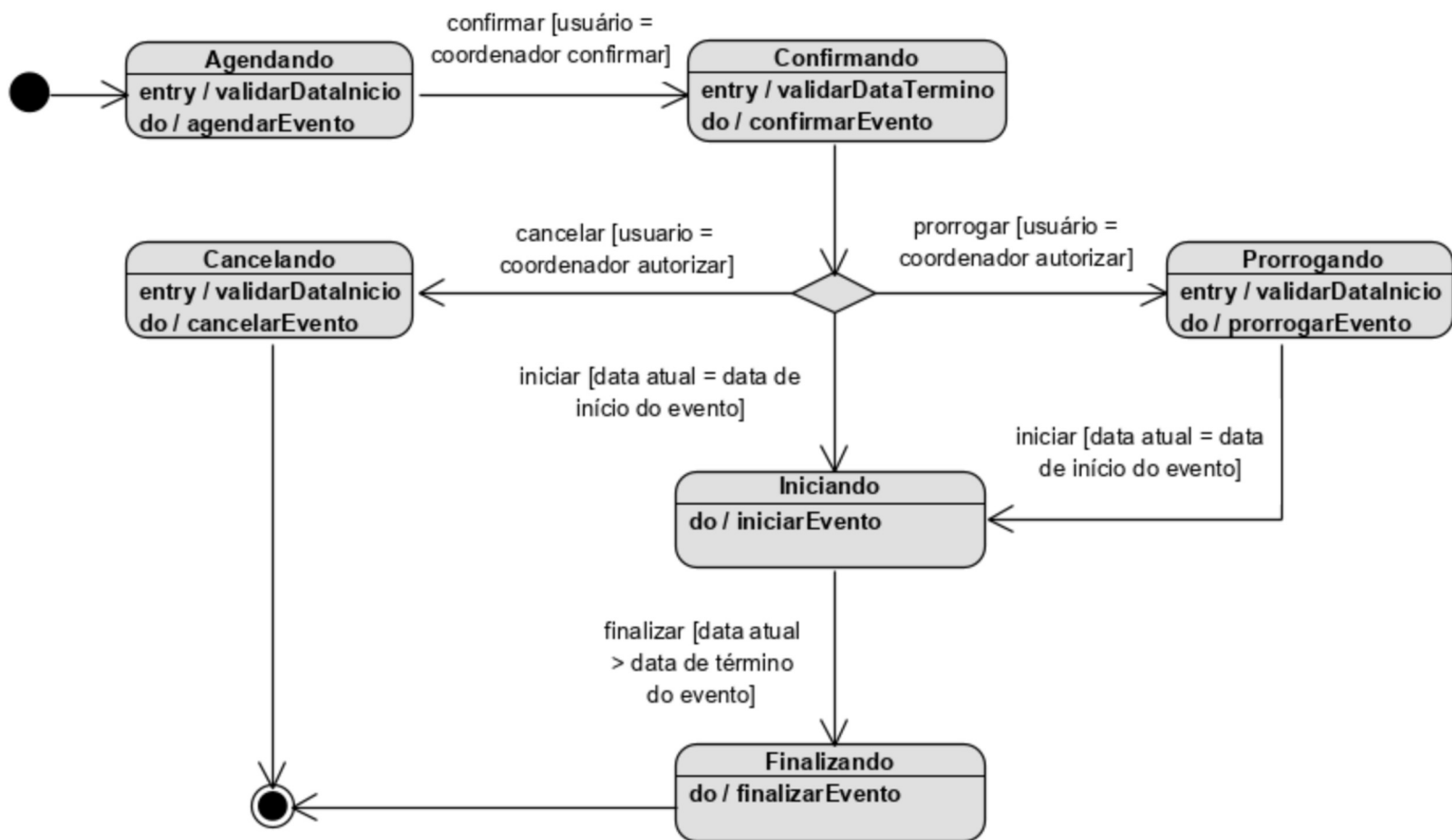


Fonte: Livro texto

Diagrama de Atividades

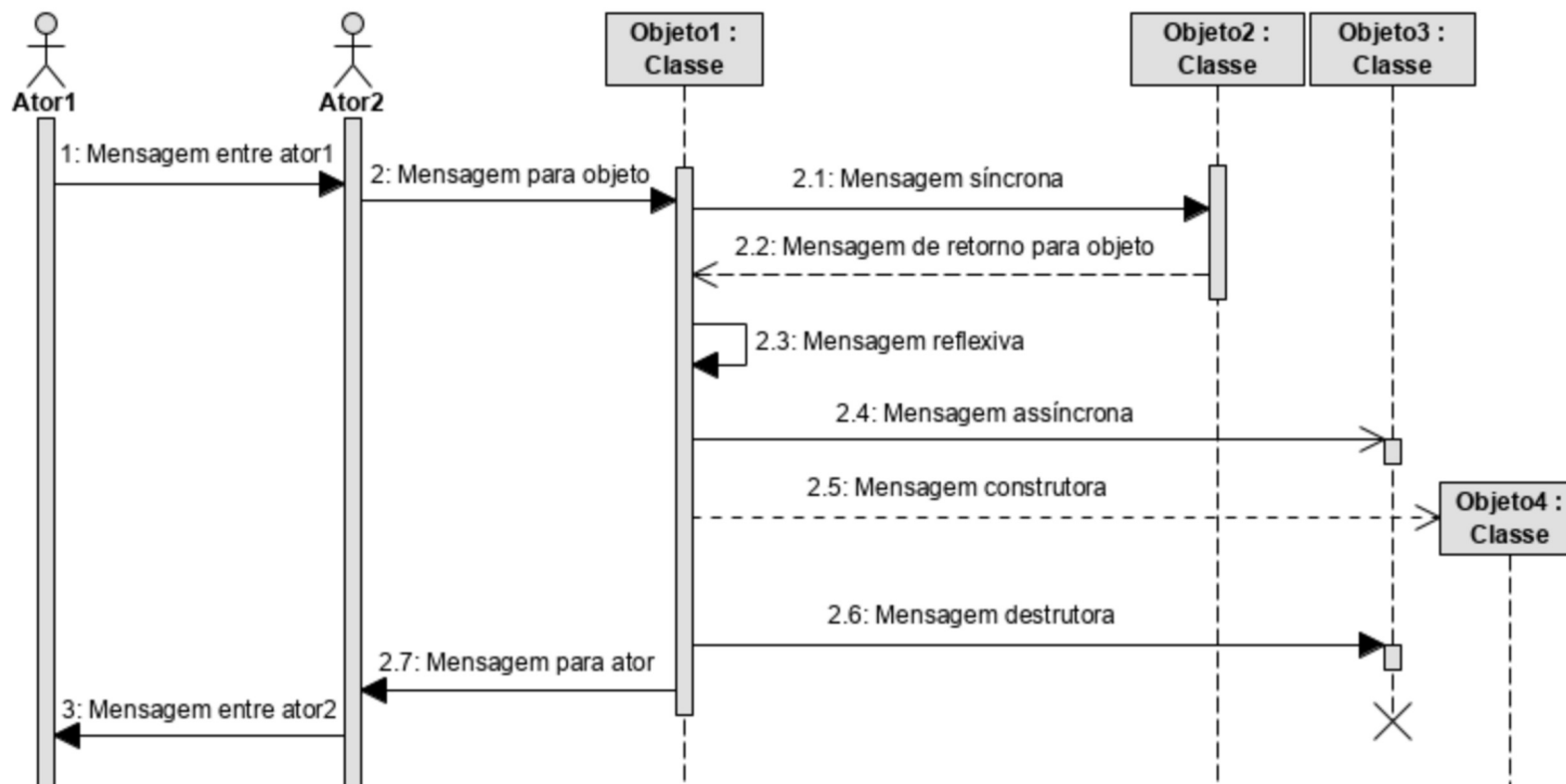


Fonte: Werlich (2020)



Fonte: Catarino (2020)

Diagrama de Sequência



Fonte: Catarino (2020)

Processo Unificado

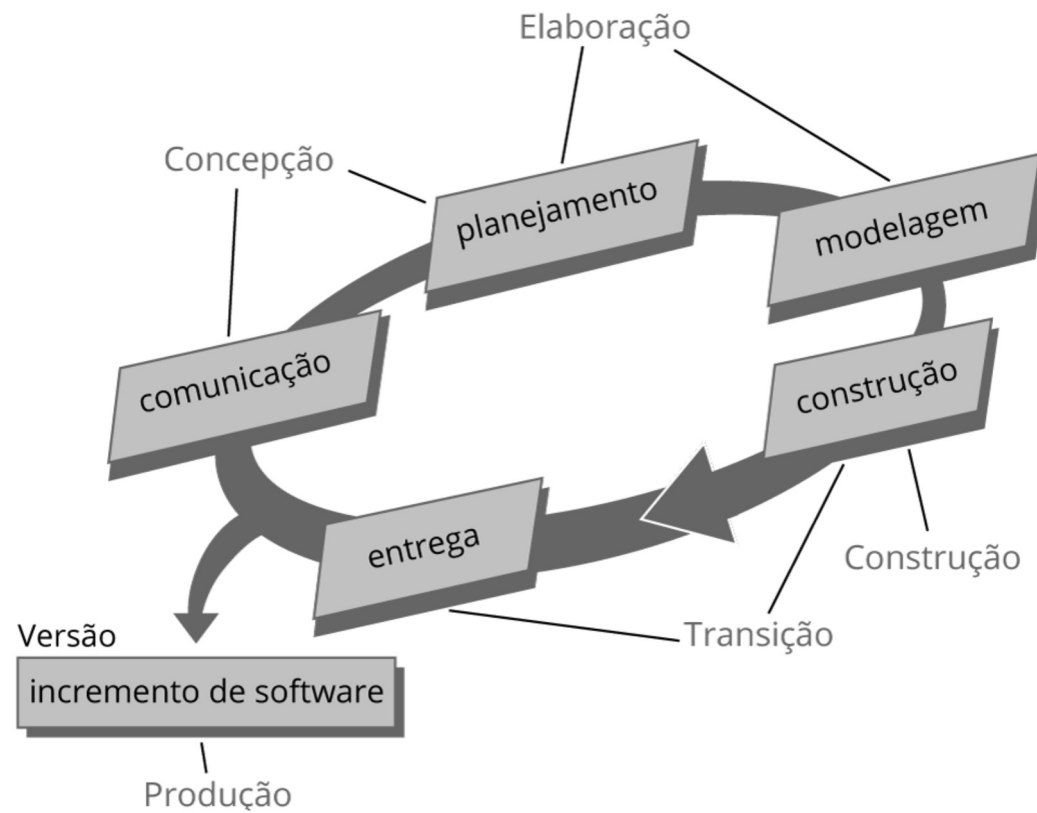
Processo Unificado

- Processo —→ quem, o que, como e quando;
- Processo Unificado (PU);
- RUP (do inglês *Rational Unified Process*);

Processo Unificado

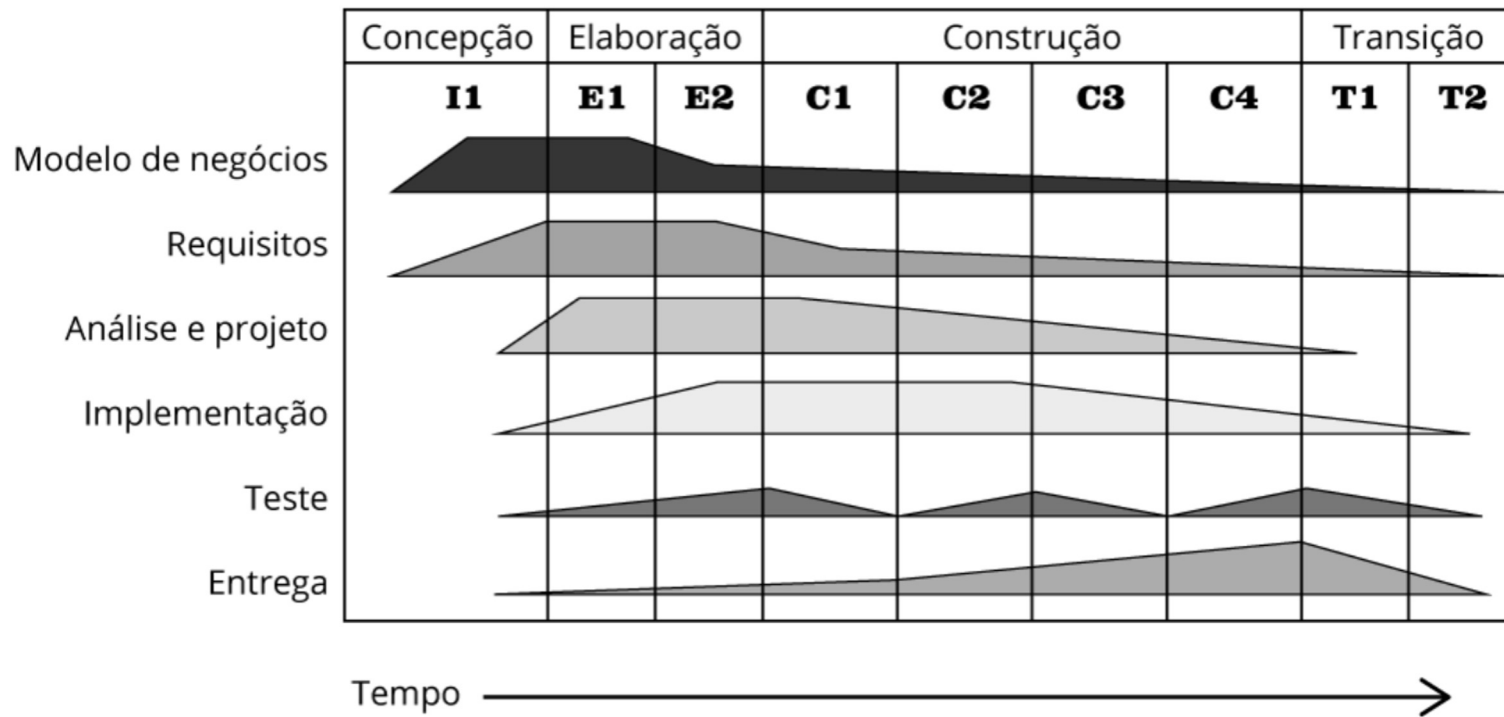
- Interativo e incremental;
- Dirigido por uma lista de casos de uso;
- Focado na arquitetura do sistema;
- Orientado a riscos;

Fases do processo unificado



Fonte: Livro texto

Fluxo de trabalho



Fonte: Livro texto

Fase de Concepção

- Diagrama de Caso de Uso;
- Diagramas de sequência;
- Diagrama de colaboração;
- Diagrama de atividades;
- Diagrama de máquinas de estado;

Fase de elaboração

- Diagrama de classes;
- Diagramas de sequência;
- Diagrama de colaboração;
- Diagrama de atividades;
- Diagrama de máquinas de estado;



Evolução

Fase de construção

- Diagrama de instalação;

Fase de Implementação

- Diagrama de classes;
- Diagramas de sequência;
- Diagrama de colaboração;
- Diagrama de atividades;

Casos de uso

Sequência

Colaboração

Atividades

Máquinas de estado

Classes

Sequência

Colaboração

Atividades

Máquinas de estado

Instalação

Transição

Sequência

Colaboração

Componentes

| | Concepção | | Elaboração | | Construção | | | | Transição | |
|--------------------|-----------|----|------------|----|------------|----|----|----|-----------|--|
| | I1 | E1 | E2 | C1 | C2 | C3 | C4 | T1 | T2 | |
| Modelo de negócios | | | | | | | | | | |
| Requisitos | | | | | | | | | | |
| Análise e projeto | | | | | | | | | | |
| Implementação | | | | | | | | | | |
| Teste | | | | | | | | | | |
| Entrega | | | | | | | | | | |

Tempo →

Tempo \longrightarrow

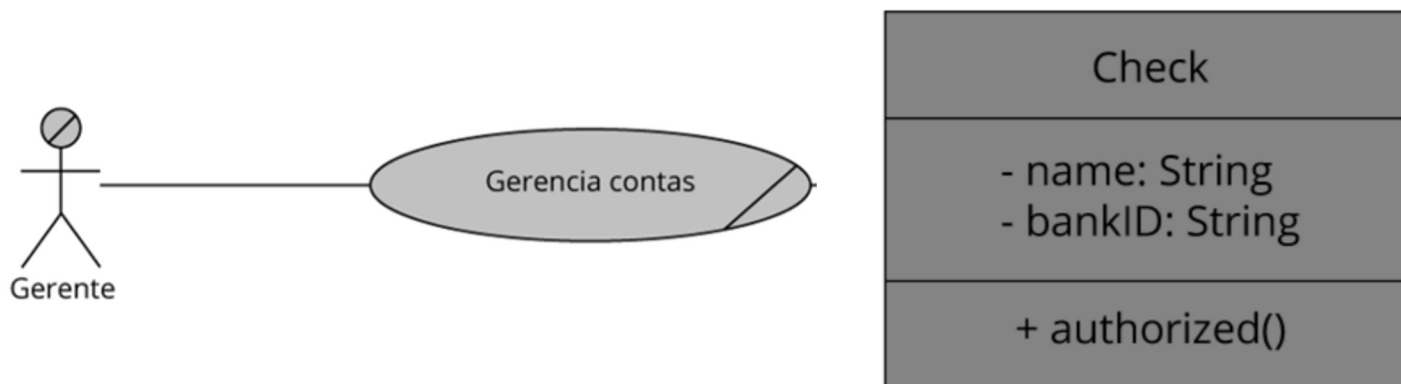
Mecanismos comuns da UML

Especificação

- Descrição exata do elemento;
- No diagrama de classe existe uma especificação;
 - Atributos;
 - Operações;
 - Comportamentos;

Adorno

- Notação gráfica dos modelos;



Fonte: Livro texto

Divisões comuns

Na modelagem de sistemas orientados a objetos, costuma ser dividido pelo menos de duas maneiras:

- Divisão de classes e objetos. Uma classe é uma abstração; um objeto é uma manifestação concreta dessa abstração.
- Separação de interface e implementação. Uma interface declara um contrato e a implementação representa uma realização completa e fiel desse contrato.

Mecanismos de extensão

- Estereótipos: é possível, na UML, utilizar o “desenho” de um determinado bloco e modificá-lo para um propósito específico, criando um novo objeto.
- Restrições: é possível, também na UML, alterar as restrições na construção de um diagrama. Em UML, as restrições são representadas pelas strings que acompanham as ligações entre elementos.

Mecanismos de extensão

- Valores predefinidos: é possível predefinir valores específicos em um diagrama, para guiar a implementação do sistema ou gerenciamento de configurações do sistema.

Regras de consistência de diagramas UML

- O número de objetos no diagrama de sequência deve ser o mesmo do número de classes;
- Deve se atentar para as atualizações do diagrama de classes e reproduzi-las corretamente no diagrama de sequência.
- O nome dos métodos deve ser respeitado entre os diagramas de classe e sequência

Regras de consistência de diagramas UML

- Os diagramas de classe e sequência devem ser sincronizados;
- Cada uma das situações representadas no diagrama de casos de uso deve ter uma operação correspondente no diagrama de classes.
- Cada caso de uso deve ter um substantivo e um verbo associados.

Regras de consistência de diagramas UML

- Para cada caso de uso deve existir ao menos um diagrama de sequência;
- Deve haver consistência entre os atores do diagrama de casos de uso e o de sequência.

Recapitulando

Recapitulando

- Características da UML;
- Diagramas UML;
 - Estruturais e comportamentais;
- Processo Unificado;
- Mecanismos comuns da UML;