


# Engenharia de Software I

Rogério Eduardo Garcia  
(rogerio.garcia@unesp.br)

*Aula 04*

*When times are stable, and the sea is calm  
and secure, no one is tested.*




unesp

Bacharelado em  
Ciência da  
Computação  
2025

1

## Engenharia de Software I – Aula 4

- Revisão
- Introdução ao Método Larman
  - Planejar e Elaborar
  - Construir
    - Analisar
- Revisão de conceitos de Orientação a Objetos



unesp  
BCC  
2025

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

3

3

# Contextualizando... ISO 12207: Estrutura

Processos Fundamentais

Aquisição

Fornecimento

Desenvolvimento

Operação

Manutenção

Processos de Apoio

Documentação

Gerenciamento de Configuração

Garantia de Qualidade

Verificação

Validação

Revisão Conjunta

Auditoria

Resolução de Problemas

Processos Organizacionais

Gerência

Melhoria

Infra-estrutura

Treinamento

Adaptação

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

4

4

# Análise de Requisitos

Engenharia de  
Sistemas de  
Computador

ANÁLISE DE  
REQUISITOS

Projeto de  
Software

Como Proceder?

Processo de Descoberta e Refinamentos

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

10

10

Prof. Dr. Rogério E. Garcia

2

## Princípios da Análise



- O domínio de informação de um problema deve ser representado e compreendido
- Modelos que descrevam a informação, função e comportamento do sistema devem ser desenvolvidos
- Os modelos devem ser divididos em partições, de maneira que revele os detalhes em forma de camadas, preferencialmente
- O processo de análise deve ter como foco a informação essencial do (Udel) – detalhes de implementação ficam para a fase de Projeto

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

11

11

## Princípios da Especificação



- Separar funcionalidade de implementação
- Uso de uma linguagem de especificação orientada ao processo
- A especificação deve abranger o sistema do qual o software é um componente
- A especificação deve abranger o sistema no qual o software opera

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

12

12

## Princípios da Especificação (Cont.)



- Uma especificação deve ser um modelo cognitivo
- Uma especificação deve ser operacional
- Uma especificação deve ser tolerante com a não-inteireza e ser expansível
- Uma especificação deve ser localizada e fracamente acoplada

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

13

13

## Para que tudo isso?



- Obter uma descrição dos requisitos
- Propor uma solução (“software”) que atenda ao requisitos da melhor maneira possível
- Possibilidade de avaliar não apenas a proposta, mas também as consequências de decisões tomadas em tempo de projeto

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

14

14

# Problema X Solução

The diagram is a 2x2 matrix with 'Problema' and 'Solução' as columns and 'Abstrato' and 'Concreto' as rows. A vertical blue arrow points upwards from 'Concreto' to 'Abstrato' in the 'Problema' column. A horizontal blue arrow points from 'Abstrato' in the 'Problema' column to 'Abstrato' in the 'Solução' column. A vertical blue arrow points downwards from 'Abstrato' to 'Concreto' in the 'Solução' column.

unesp  
BCC  
2025

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

15

15

# Portanto...

- É preciso planejar considerando a maneira com que um projeto será implementado...

**Orientado a Objetos**

unesp  
BCC  
2025

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

16

16

## Mundo x Paradigma OO



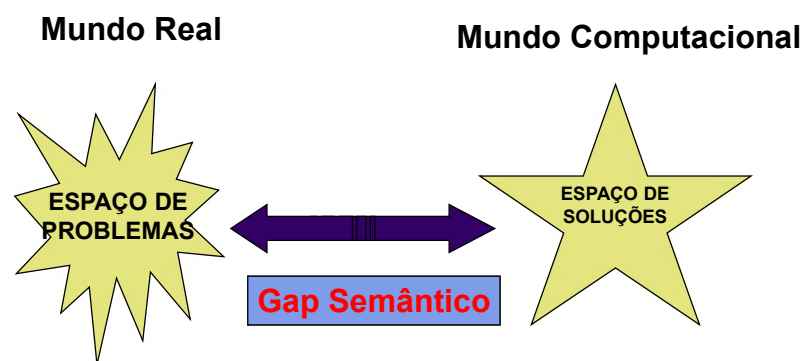
- Mundo Real é formado por objetos que se interagem
- Representar esses objetos em um software é mais natural e permanente do que representar a sua funcionalidade (decomposição funcional), pois essa é mutável
- A representação usando objetos facilita o mapeamento do mundo real, ou seja, a criação de um modelo que o represente

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

17

17

## Problema...



Todo software representa um **Modelo** de um problema do mundo real, no Espaço de Soluções

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

18

18

## Gap Semântico



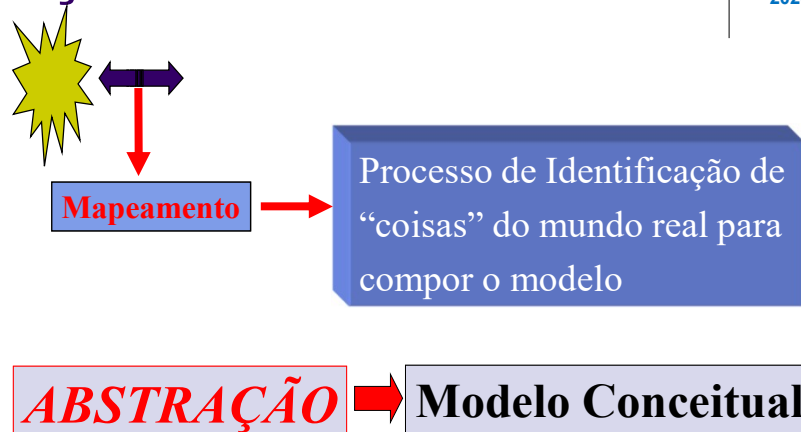
- Quanto menor o Gap (diferença entre espaços) mais fácil será :
  - o desenvolvimento da aplicação
  - assegurar a compreensão, confiabilidade e manutenção da aplicação
- Diminuir o Gap implica em tornar o mapeamento do mundo real (modelo) mais próximo da realidade
- Sendo mais “natural”, o Paradigma de Orientação a Objetos tem por objetivo diminuir o gap semântico

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

19

19

## Criação de Modelos



Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

20

20

unesp  
BCC  
2025

# Problema X Solução

The diagram is a 2x2 matrix with 'Problema' and 'Solução' as columns and 'Abstrato' and 'Concreto' as rows. A vertical blue arrow on the left points upwards from 'Concreto' to 'Abstrato'. A horizontal blue arrow at the top points from 'Problema' to 'Solução'. A vertical blue arrow on the right points downwards from 'Abstrato' to 'Concreto'.

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

21

21

unesp  
BCC  
2025

# Método Larman

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

22

22



# Método Larman: Visão Geral

The diagram illustrates the components of the Larman Method. A central light blue rectangle labeled "Tópicos e Habilidades" is surrounded by six yellow ovals: "A/POO" (top), "Padrões" (top-left), "Princípios e Diretrizes" (bottom-left), "Desenvolvimento Iterativo" (bottom), "Engenharia de Requisitos" (bottom-right), and "Notação UML" (top-right).

unesp  
BCC  
2025

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

23

23

The logo for the Unified Modeling Language (UML) features the text "UNIFIED MODELING LANGUAGE" in a purple sans-serif font, followed by a large, stylized "UML" symbol. The "U" is purple, the "M" is yellow, and the "L" is purple. A small "TM" trademark symbol is to the right of the "L".

unesp  
BCC  
2025

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

24

24

# UML



- História da UML

- início em 1994 – esforço conjunto de Booch e Rumbaugh para combinar as notações diagramáticas de seus métodos Booch e OMT (*Object Modeling Technique*)
- a eles juntaram-se outros colaboradores
- adotada como padrão em 1997 pela OMG
- continua a ser refinada...
  - versão 2.0 em andamento (???)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

25

25

# UML



- Segundo OMG: “A UML (Unified Modeling Language - Linguagem de Modelagem Unificada) é uma linguagem para especificar, visualizar, construir e documentar os artefatos de sistemas de software, bem como para modelar negócios e outros sistemas que não sejam de software”
- Notação UML - principalmente diagramática, para modelagem de sistemas usando conceitos baseados na metáfora de “objetos”

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

26

26

## UML



- Questões críticas em análise e projeto OO:
  - Como alocar responsabilidades às classes?
  - Com quem os objetos devem interagir?
  - O que cada classe deve fazer?
- Certas soluções que já se mostraram boas para problemas de projeto podem ser expressas como conjuntos de princípios, heurísticas e padrões

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

27

27

## UML



- A UML combina o melhor do melhor de:
  - Conceitos de Modelagem de Dados (Diagramas Entidade-Relacionamento);
  - Modelagem de Negócios (Fluxo de trabalhos);
  - Modelagem de Objetos;
  - Modelagem de Componentes.

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

28

28

## UML



- A UML é a linguagem padrão para visualizar, especificar, construir e documentar os artefatos de um sistema intensamente baseado em software
- Pode ser usada com todos os processos, durante todo o ciclo de desenvolvimento, e com diferentes tecnologias de implementação;

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

29

29

## Diagramas

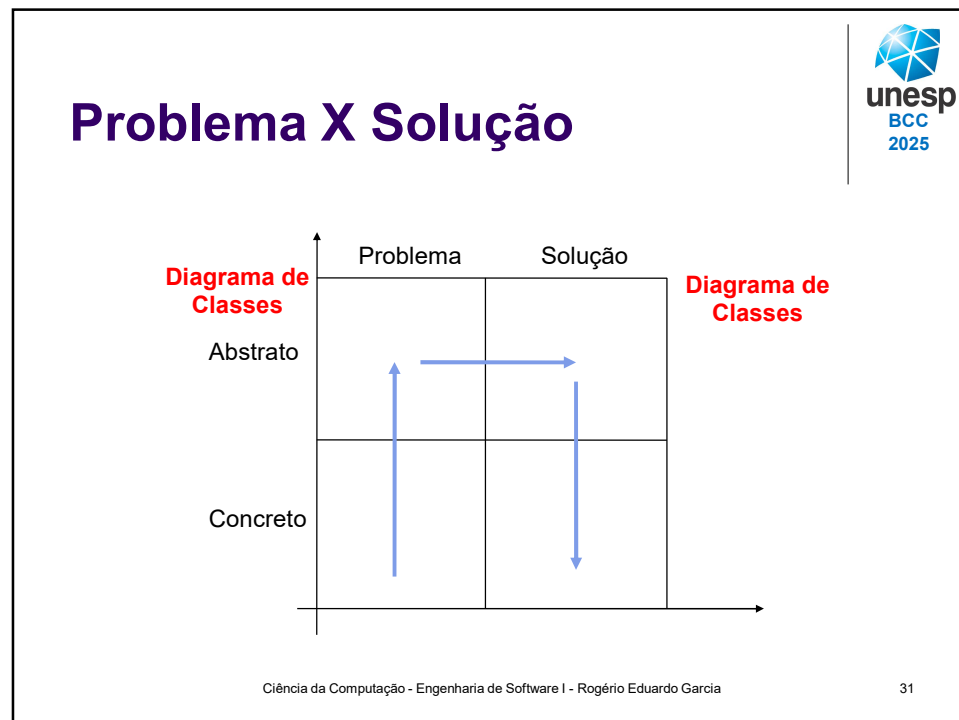


- Use Case
- Diagramas de Estrutura Estática
  - Diagrama de Objetos
  - Diagramas de Classe
- Diagramas de Interação
  - Diagrama de Sequência
  - Diagrama de Colaboração
- *Statecharts*
- Diagramas de Atividade
- Diagrama de Implementação
  - Diagrama de Componentes
  - Diagrama de Desdobramentos (*Deployment*)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

30

30



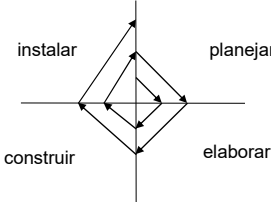
31




32

## Desenvolvimento Iterativo

- Um ciclo de vida iterativo (CVI) envolve a repetição dos ciclos de planejamento, elaboração, construção e instalação
- O sistema cresce pela **adição** de novas funções (e **refinamento** das existentes) em cada ciclo iterativo
- Cada ciclo ataca um pequeno conjunto de requisitos



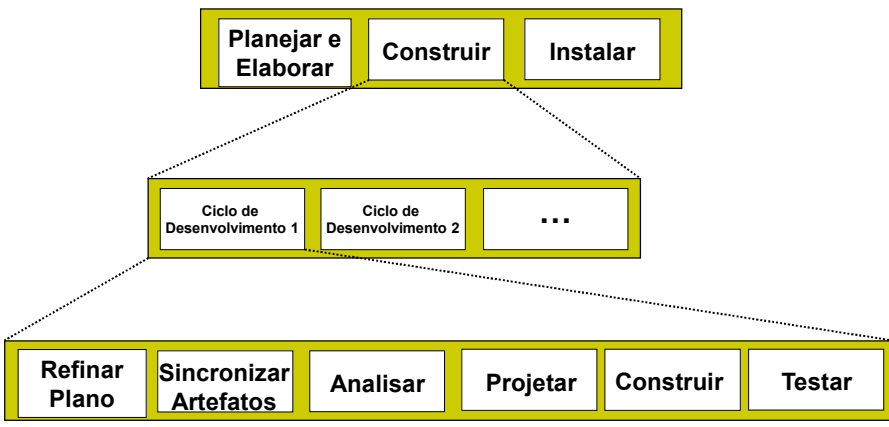
Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia




unesp  
BCC  
2025

33

## Método Larman



Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia



unesp  
BCC  
2025

34

unesp

BCC

2025

# Método Larman

Planejar e Elaborar

Construir

Instalar

Ciclo de Desenvolvimento 1

Ciclo de Desenvolvimento 2

...

Refinar Plano

Sincronizar Artefatos

Analisar

Projetar

Construir

Testar

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

35

35

unesp

BCC

2025

# Para que isso?

Projeto

Versus

Código Fonte

Projeto & Código Fonte

28/03/2025

Prof. Dr. Rogério Eduardo Garcia

36

36

unesp  
BCC  
2025

# Engenharia de Software?

Para que E.S.?

The diagram illustrates the flow of information in software engineering. On the left, a box labeled 'Atores' (Actors) contains three smiley faces. A dashed line connects this box to a central blue circle labeled 'Projeto' (Project). A green arrow labeled 'Informação' (Information) points from the 'Projeto' to a box labeled 'Programadores' (Programmers), which also contains three smiley faces. A dashed line also connects the 'Atores' box directly to the 'Programadores' box. To the right of the 'Programadores' box is a stack of blue rectangles labeled 'Código Fonte' (Source Code). A red box labeled 'Engenharia de Software' (Software Engineering) is positioned above the 'Código Fonte' stack. The text 'Prof. Dr. Rogério Eduardo Garcia' is written below the 'Programadores' box.

28/03/2025

37

37

unesp  
BCC  
2025

# Engenharia de Software?

Para que E.S.?

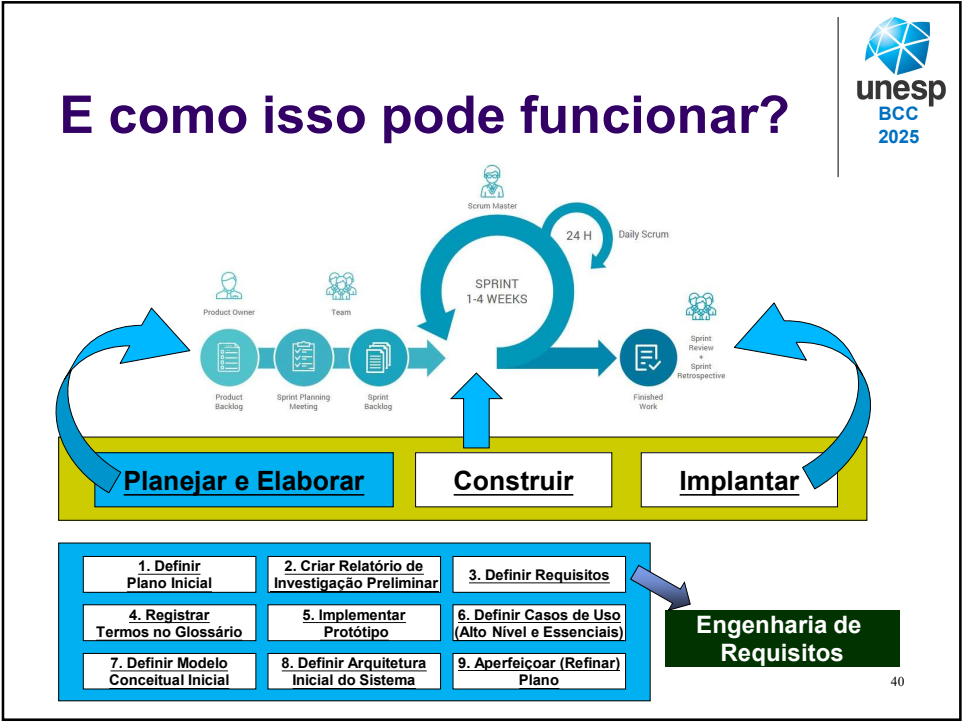
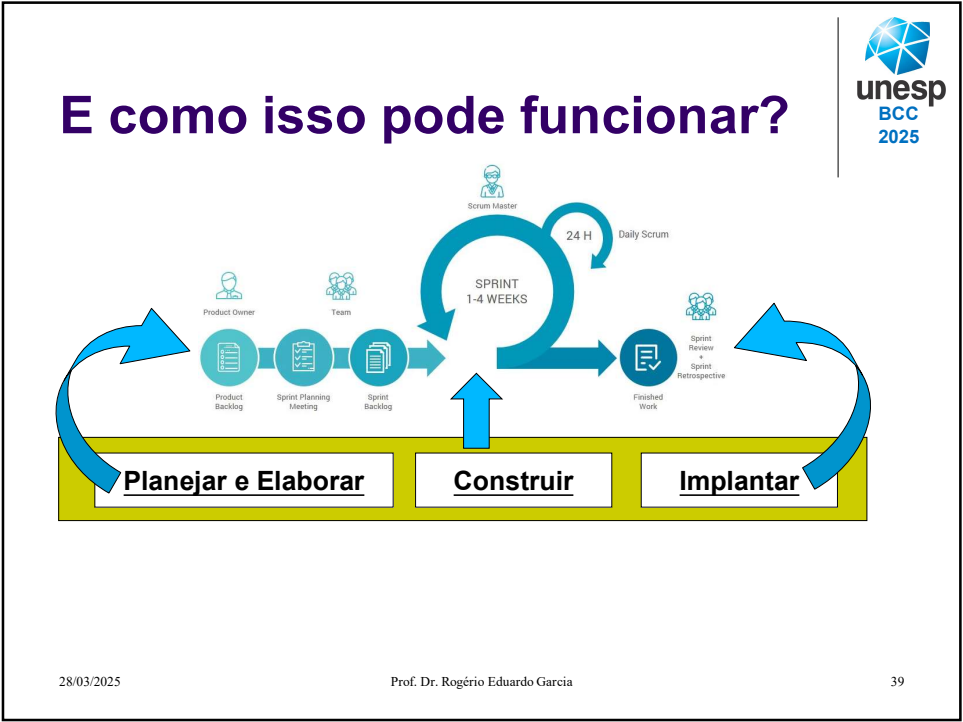
This diagram is similar to the one on slide 37, but it includes three yellow starburst shapes labeled 'Qualidade' (Quality). One starburst is positioned above the 'Projeto' circle, another is above the 'Engenharia de Software' box, and the third is above the 'Código Fonte' stack. The rest of the diagram, including the 'Atores', 'Programadores', and 'Informação' flow, remains the same. The text 'Prof. Dr. Rogério Eduardo Garcia' is written below the 'Programadores' box.

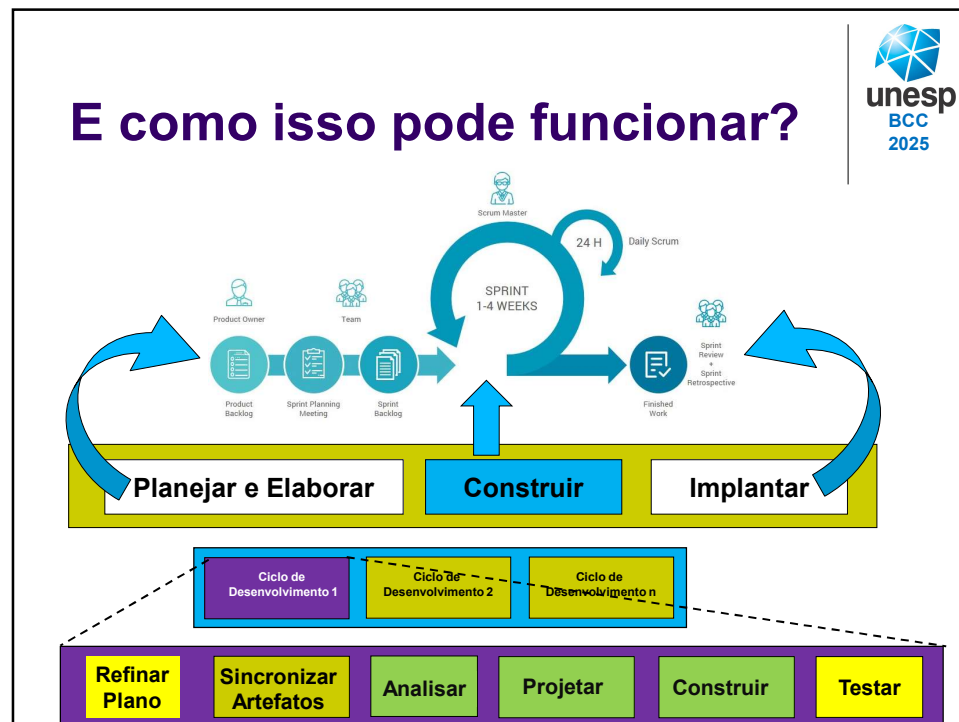
28/03/2025

38

38







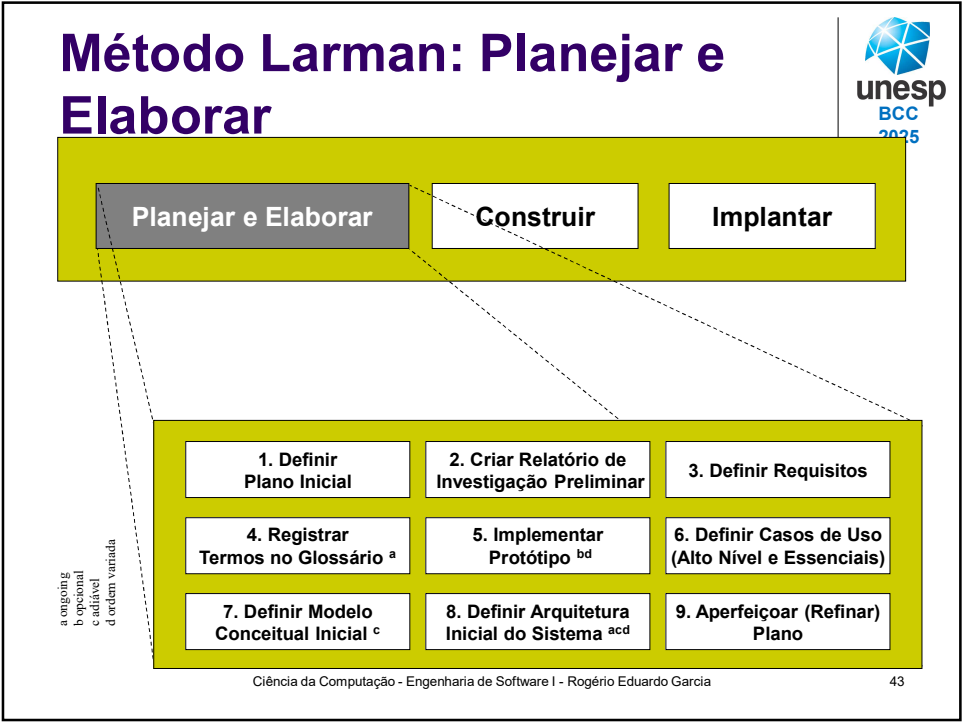
41

## Método Larman: Visão Geral

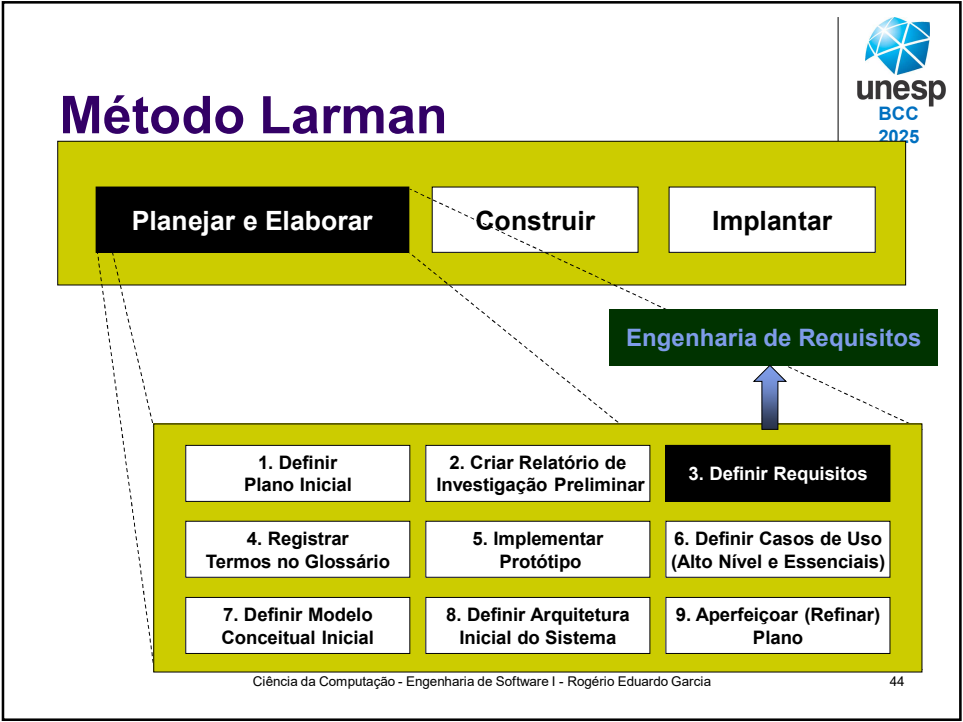
- Processo **Planejar e Elaborar**
  - Casos de Uso: Formatos, Tipos e Diagrama
  - Modelo Conceitual: Conceitos e Associações
- Processo **Construir** (Fase **Analisar**)
  - Casos de Uso: refinar e evoluir (descrição e diagrama)
  - Modelo Conceitual: Agregações, Generalizações e Tipos Associativos
  - Diagramas de Sequência
  - Contratos de Operação

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

42



43



44

## Estudo de caso - TPV



- Descrição Geral:
  - O propósito deste projeto é criar um terminal de ponto de vendas (TPV) para ser usado em lojas de varejo
- Clientes
  - ObjectStore, Inc., uma multinacional que comercializa objetos

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

45

45

## Estudo de caso - TPV



- Objetivos:
  - O objetivo geral é aumentar a automatização das compras (*checkout*) para permitir serviços e processos comerciais mais rápidos, melhores e mais baratos. Tipicamente, isso inclui:
    - *checkout* (passagem pelo caixa) mais rápido para o cliente
    - análise rápida e precisa do crédito
    - controle automático do estoque

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

46

46

## TPV - Funções Básicas



- R1.1 – Registrar a venda em andamento (corrente), isto é, os itens comprados (E)
- R1.2 – Calcular o total da venda corrente, incluindo os cálculos de impostos e de cupons de desconto (E)
- R1.3 – Capturar a informação de um item adquirido, usando o código, obtido por um leitor de código de barra, ou pela entrada manual do código do produto, usando o código universal de produto (CUP ou UPC) (E)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

47

47

## Funções Básicas - TPV



- R1.4 – Reduzir a quantidade em estoque quando a venda for finalizada (O)
- R1.5 – Registrar as vendas completadas (O)
- R1.6 – O funcionário (Caixa) deve abrir o caixa (log in) com um Identificador (ID) e uma senha para poder usar o sistema (E)
- R1.7 – Fornecer um mecanismo de armazenamento permanente (O)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

48

48

## Funções Básicas - TPV



- R1.8 – Fornecer mecanismos de comunicação inter-processos e inter-sistemas (O)
- R1.9 – Exibir a descrição e o preço do item registrado (E)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

49

49

## Funções de Pagamento - TPV



- R2.1 – Tratar os pagamentos em dinheiro: capturar a quantia recebida e informar o troco (E)
- R2.2 – Tratar o pagamento com cartão de crédito: captar a informação do cartão de crédito por um leitor de cartões ou uma entrada manual e autorizar o pagamento com o serviço de autorização de crédito (externo) da loja via conexão por modem (E)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

50

50

## Funções de Pagamento - TPV



- R2.3 – Registrar os pagamentos por crédito no sistema de contas a receber da loja, uma vez que o serviço de autorização de crédito deve à loja a quantia oferecida como pagamento (O)
- R2.4 – Tratar os pagamentos com cheque: capturar o CPF por entrada manual e autorizar o pagamento com o serviço de autorização de crédito da loja (externo) via conexão por modem (E)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

51

51

## Atributos do Sistema – TPV



- para R1.9 (Exibir a descrição e o preço do item registrado (E))
  - tempo de resposta: Max 5s → Obrigatório
  - metáfora da interface:
    - saída baseada em formulário → Obrigatório
    - saída colorida → Desejável


Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

52

52

## Atributos do Sistema – TPV

- para R2.3 (Registrar os pagamentos por crédito no sistema de contas a receber da loja (O))
  - tolerância a falhas: deve registrar no sistema de contas a receber em 24h, mesmo em caso de falhas elétrica ou de hardware → Obrigatório
  - tempo de resposta: Max 10s → Obrigatório

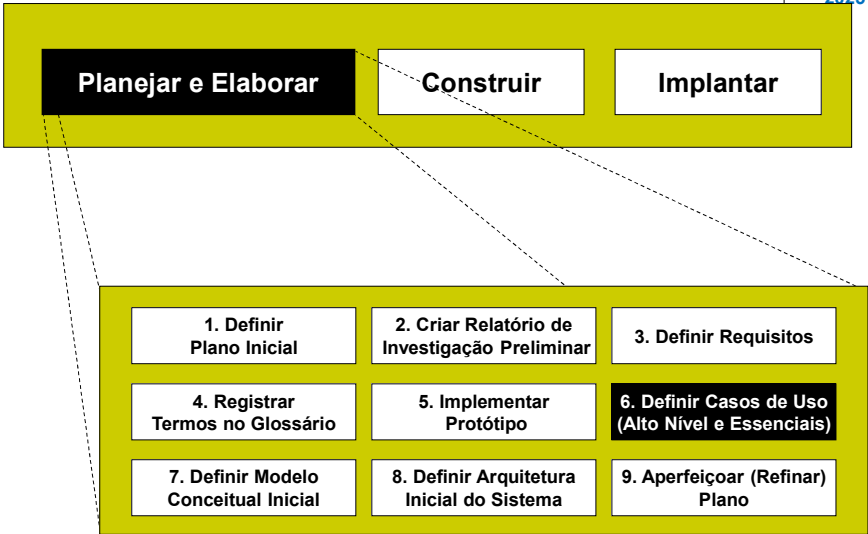



Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

53

53

## Método Larman





Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

54

54



## Casos de Uso: descrevendo processos



- Objetivos:
  - Identificar e Escrever Casos de Uso
  - Elaborar Diagramas de Casos de Uso
- Mais adiante (*Análise do Construir*) será usado para:
  - Contrastar Casos de Uso de Alto Nível e Expandidos
  - Contrastar Casos de Uso Reais e Essenciais

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

55

55

## Casos de Uso



- Um **caso de uso** é um padrão de comportamento que o sistema exhibe
  - Cada caso de uso é uma sequência de transações relacionadas executadas por um **ator** e o sistema em um diálogo

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

56

56

## Casos de Uso



- Um **caso de uso** é um documento textual que descreve a sequência de eventos realizados por um **ator** (um agente externo) para completar um processo durante o uso do sistema
- Contam “histórias” de utilização do sistema
- Casos de uso **não** são especificação de requisitos, mas ilustram e implicam requisitos
  - dependem de que se tenha um entendimento ao menos parcial dos requisitos do sistema

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

57

57

## Atores



- Um ator é uma **entidade externa** ao sistema que participa de um caso de uso de alguma forma.
- Atores interagem com o sistema, estimulando-o com eventos de entrada ou de saída
- Representam o **papel** que desempenham no caso de uso. Ex: Cliente, Caixa
  - uma pessoa, por exemplo, pode assumir vários papéis
  - várias pessoas podem ser instâncias de um ator

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

58

58

## Atores



- Atores podem ser papéis desempenhados por pessoas, sistemas de computadores, dispositivos elétricos e mecânicos, ...
- Para um caso de uso, geralmente existe um ator iniciador e possivelmente vários outros atores participantes

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

59

59

## Método Larman: Formatos dos Casos de Uso



- de alto nível
  - descreve o processo sucintamente, em duas ou três sentenças
  - são vagos a respeito de decisões de projeto e são úteis para a compreensão dos principais processos globais
- expandidos
  - mostram mais detalhes
  - compreensão mais profunda dos processos e requisitos

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

60

60




## Caso de Uso de Alto Nível

<b>Caso de uso:</b>	<b>Comprar Itens</b>	Usar verbo para nomear caso de uso
<b>Atores:</b>	Cliente, Caixa	Nome de atores com letra maiúscula
<b>Tipo:</b>	primário (a ser discutido adiante...)	
<b>Descrição:</b>	Um Cliente chega ao balcão de saída da loja com itens que deseja comprar. O Caixa registra os itens de compra e recebe o pagamento. Quando termina, o Cliente sai com os itens comprados.	

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

61

61



## Expandindo Casos de Uso

- Um documento de fluxo de eventos é criado para cada caso de uso
  - Escrito do ponto de vista do ator
- Detalha o que o sistema deve fornecer quando o caso de uso é executado
- Conteúdos típicos
  - Como o caso de uso inicia e termina
  - Fluxo normal de eventos
  - Fluxos alternativos de eventos
  - Fluxos excepcionais de eventos (respostas a erros)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

62

62

# Caso de Uso Expandido

Parte 1

Descrição similar ao  
Caso de uso de Alto Nível

Parte 2

Descrição da sequência  
típica de eventos

Parte 3

Descrição de sequências  
Alternativas de eventos

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

63

63

# Caso de Uso Expandido (Parte 1 - Resumo)

unesp  
BCC  
2025

(restrito a pagamento em dinheiro e sem tratar controle de estoque)

Caso de Uso: **Comprar Itens com Dinheiro**

Atores:	Cliente (iniciador), Caixa	Informar ator que inicia o processo
Finalidade:	Capturar a venda e seu pagamento em dinheiro	
Visão geral:	Um Cliente chega ao balcão de saída da loja com itens que deseja comprar. O Caixa registra os itens de compra e recebe o pagamento. Quando termina, o Cliente sai com os itens comprados.	
Tipo:	primário e essencial (a ser discutido adiante...)	
Referências Cruzadas:	Requisitos: R1.1, R1.2, R1.3, R1.7, R1.9, R2.1	

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

64

64

## Caso de Uso Expandido



- Rastreabilidade
  - A cláusula de **referência cruzada** permite conferir se todos os requisitos foram atendidos por casos de uso.
  - Ao final, todos os casos de uso devem poder ser rastreados para a implementação e o teste.

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

65

65

## Lembrando...



- R1.1 – Registrar a venda em andamento ... (E)
- R1.2 – Calcular o total da venda corrente ... (E)
- R1.3 – Capturar a informação de um item adquirido, usando o código... (E)
- R1.7 – Fornecer um mecanismo de armazenamento permanente (O)
- R1.9 – Exibir a descrição e o preço do item registrado (E)
- R2.1 – Tratar os pagamentos em dinheiro... (E)


Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

66

66

## Caso de Uso Expandido

### Parte 2 - Sequência típica de eventos



Ação do ator	Resposta do Sistema
1. Este caso de uso começa quando o Cliente chega ao TPV com itens para comprar	
2. O Caixa registra o identificador de cada item  Se há mais de um do mesmo item, o caixa também entra a quantidade	3. Determina o preço do item e adiciona informação sobre o item à transação de venda corrente  A descrição e o preço do item são apresentados
4. Quando termina a entrada dos itens, o Caixa indica ao TPV que as entradas estão completas	5. Calcula e apresenta o total da venda
6. O Caixa informa o total ao cliente	
7. O Cliente entrega o pagamento em dinheiro – o “pagamento em dinheiro” – possivelmente maior que o total da venda	


Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

67

67

## Caso de Uso Expandido

### Parte 2 - Sequência típica de eventos-Cont.



Ação do ator	Resposta do Sistema
8. O Caixa registra a quantidade de dinheiro recebida	9. Exibe o valor do troco a ser devolvido ao cliente
10. O Caixa deposita o dinheiro recebido e retira o troco devido  O Caixa entrega ao cliente o troco e o recibo impresso	11. Registra a venda completada ( <i>logs</i> )
12. O Cliente sai com os itens comprados	

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

68

68

## Caso de Uso Expandido (Parte 3 – Sequências Alternativas)



- Descreve alternativas importantes ou exceções que podem ocorrer numa sequência típica
  - se forem muito complexas podem se transformar num caso de uso
- Sequências alternativas:
  - Linha 2: Identificador de item inválido digitado. Indicar o erro.
  - Linha 7: O Cliente não tem dinheiro suficiente. Cancelar a transação de venda

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

69

69

## Tipos de Casos de Uso (I)



- Primários : principais processos comuns  
Ex: Comprar Itens
- Secundários: processos menos importantes ou raros  
Ex: Requisição de estoque de produto novo
- Opcionais: processos que podem não ser incluídos na solução

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

70

70



## Tipos de Casos de Uso (II)



- Essencial: caso de uso expandido expresso numa forma ideal, que é relativamente livre de detalhes tecnológicos e de implementação
  - decisões de projeto são postergadas
- Real: descreve o processo em termos de seu projeto atual (real)
  - considera tecnologia, entrada e saída, interface,...
  - definido na fase de projeto

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

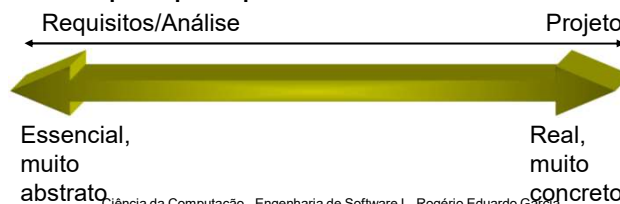
71

71

## Tipos de Casos de Uso (II)



- Casos de uso de **alto nível** são **essenciais** por natureza, devido à sua forma resumida e alto nível de abstração
- O intervalo entre essencial e real deve ser visto como um contínuo em que o caso de uso pode se situar em qualquer ponto




Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

72

72

## Caso de Uso Comprar Itens: Essencial




Ação do ator	Resposta do Sistema
1. Este caso de uso começa quando o Cliente chega ao TPV com itens para comprar	
2. O Caixa registra o identificador de cada item  Se há mais de um do mesmo item, o caixa também entra a quantidade	3. Determina o preço do item e adiciona informação sobre o item à transação de venda corrente  A descrição e o preço do item são apresentados
4. ...	....

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

73

73

## Caso de Uso Comprar Itens: Real



Ação do Ator	Resposta do Sistema
1. Este caso de uso começa quando o Cliente chega ao TPV com itens para comprar	
2. Para cada item o Caixa digita o código universal do produto no campo de entrada UPC da janela. Ele então pressiona o botão “Entrar Item” com o mouse ou pressiona <Enter>	3. Mostra o preço do item e adiciona a informação do item à transação de venda corrente. A descrição e o preço são mostrados na caixa de texto 2 da Janela1.
....	...

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

74

74

## Importante



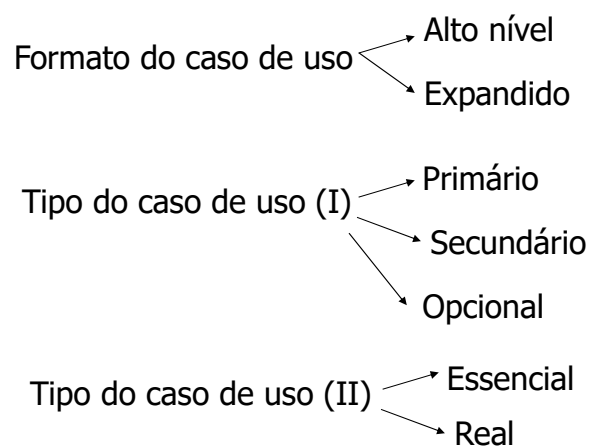
- Em geral, os casos de uso reais não devem ser produzidos na fase de engenharia de requisitos (comprometimento prematuro com uma decisão de projeto e complexidade desnecessária)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

75

75

## Resumo



Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

76

76

## Diagrama de Casos de Uso: UML Diagram



- **Objetivo**

- Mostrar como o **sistema** a ser desenvolvido irá **interagir** com o **ambiente**, delimitando o sistema e definindo a funcionalidade
- Importantes na organização e modelagem do **comportamento** do sistema

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

77

77

## Diagrama de Casos de Uso



- Um diagrama de caso de uso mostra o **relacionamento** entre os atores e os casos de uso dentro de um sistema.
  - Um **caso de uso** representa uma funcionalidade do sistema.
    - Representado por uma elipse contendo o nome do caso de uso.
  - Um **ator** é um agente externo (um usuário ou um outro sistema) que interage com o sistema.
    - Pode ser representado como um retângulo de classe com o estereótipo "ator" ou pela figura de um homem estilizado.

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

78

78

# Diagrama de Casos de Uso: Notação

The diagram illustrates the notation for a Use Case Diagram. It features two main elements: a light blue oval labeled "Comprar Itens" and a stick figure labeled "Caixa". An arrow points from the text "ícone para caso de uso" to the oval, and another arrow points from the text "ícone para ator" to the stick figure. The diagram is part of a presentation slide with a header and footer.

unesp  
BCC  
2025

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

79

79

# Diagrama de Casos de Uso TPV

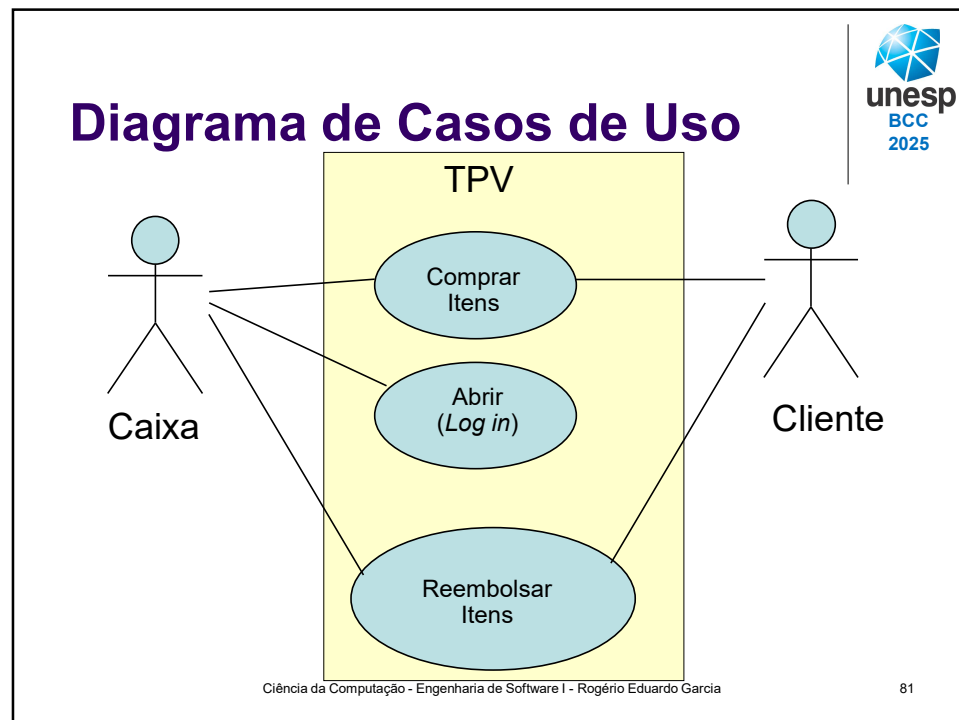
The diagram shows a Use Case Diagram for a TPV (Terminal Point of Sale) system. It includes two actors, "Caixa" and "Cliente", and three use cases: "Comprar Itens", "Abrir (Log in)", and "Reembolsar Itens". "Caixa" is connected to "Comprar Itens", "Abrir (Log in)", and "Reembolsar Itens". "Cliente" is connected to "Comprar Itens" and "Reembolsar Itens". The diagram is part of a presentation slide with a header and footer.

unesp  
BCC  
2025

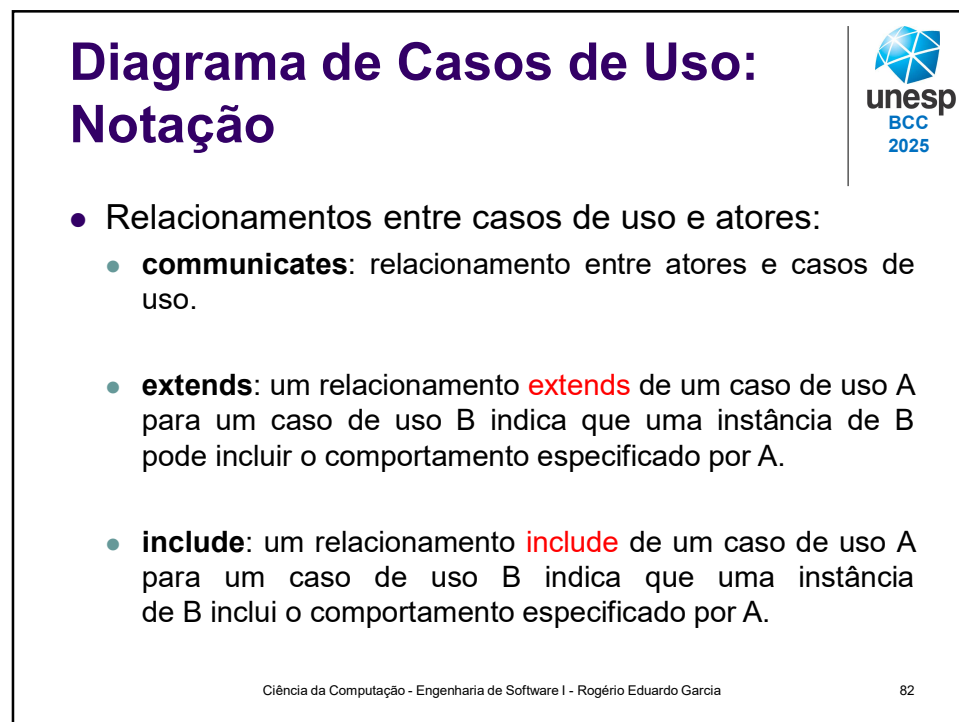
Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

80

80



81



82

# Diagrama de Casos de Uso: Notação

The diagram illustrates two types of use case relationships: **Relacionamento <<include>>** and **Relacionamento <<extends>>**.

**Relacionamento <<include>>**: A stick figure labeled 'Cliente' is connected by a solid arrow to an oval labeled 'RealizarPedido'. A dashed arrow points from 'RealizarPedido' to another oval labeled 'ValidarCliente', with the label '<<include>>' placed next to the dashed arrow.

**Relacionamento <<extends>>**: A stick figure labeled 'Cliente' is connected by a solid arrow to an oval labeled 'RealizarPedido'. A dashed arrow points from an oval labeled 'CadastrarCliente' to 'RealizarPedido', with the label '<<extends>>' placed next to the dashed arrow.

83

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

# Diagrama de Casos de Uso: Exemplo

The diagram shows a 'Sistema de Biblioteca' (Library System) with several use cases and actors.

**Actors:** 'Leitor' (Reader) and 'Bibliotecário' (Librarian).

**Use Cases:** 'Reservar', 'Cancelar reserva', 'Emprestar', 'Devolver', 'Adicionar título', 'Remover/atualizar título', 'Adicionar item', 'Adicionar leitor', and 'Manter'.

**Relationships:**

- 'Leitor' is connected to 'Reservar', 'Cancelar reserva', 'Emprestar', and 'Devolver'.
- 'Bibliotecário' is connected to 'Adicionar título', 'Remover/atualizar título', 'Adicionar item', 'Adicionar leitor', and 'Manter'.
- 'Manter' is connected to 'Adicionar título', 'Remover/atualizar título', 'Adicionar item', and 'Adicionar leitor'.
- 'Cancelar reserva' is connected to 'Emprestar' with the label '<<extends>>'.
- 'Adicionar título' is connected to 'Manter' with the label '<<extends>>'.
- 'Remover/atualizar título' is connected to 'Manter' with the label '<<extends>>'.
- 'Adicionar item' is connected to 'Manter' with the label '<<extends>>'.
- 'Adicionar leitor' is connected to 'Manter' with the label '<<extends>>'.

84

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

## Identificação de casos de uso



- Dois métodos: baseado nos atores ou baseado nos eventos do sistema
- **Baseado em atores**
  - Identificar os atores relacionados a um sistema ou organização
  - Para cada ator, identificar os processos que eles iniciam ou dos quais eles participam
  - Exemplos:
    - Caixa - Iniciar uso, Registrar retirada de dinheiro
    - Cliente – Comprar itens, Reembolsar itens

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

85

85

## Identificação de casos de uso



- **Baseado em eventos**
  - Identificar os eventos externos aos quais um sistema deve responder
  - Relacionar os eventos a atores e a casos de uso
  - Exemplos:
    - Itens vendidos (ator=cliente, caso de uso=comprar item)
    - Dinheiro retirado (ator=caixa, caso de uso=registrar retirada de dinheiro)

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

86

86



## Importante



- um caso de uso não representa um passo individual ou uma operação ou transação de entrada. por exemplo: “imprimir o recibo” não é um caso de uso no sistema de TPV
- um caso de uso é normalmente a descrição de um processo relativamente grande, com início e fim próprios, que normalmente incluem várias transações ou operações de entrada e saída. Ex:
  - retirar dinheiro de um caixa automático
  - matricular-se em uma disciplina
  - verificar ortografia em um editor de texto
  - ...

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

87

87

## Escopo do Sistema



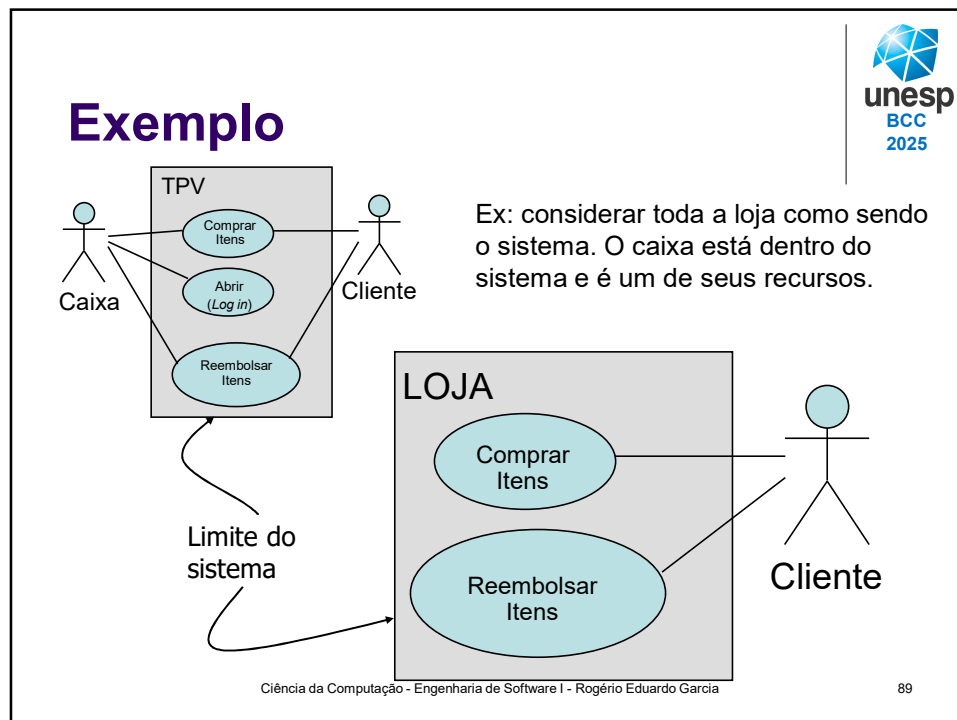
- O sistema pode ser limitado por:
  - Hardware ou software
  - Departamentos de uma organização
  - Toda a organização
- O limite é sempre delimitado arbitrariamente pelo analista e o cliente, mas geralmente leva em conta critérios tais como: política organizacional, limites de menor comunicação entre os subsistemas, oportunidade e tamanho do sistema

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

88

88



89

## Decisão e Desvio

- Pontos de decisão e desvio podem ocorrer em um caso de uso
  - Ex: no caso de uso Comprar Itens, o cliente pode pagar em dinheiro, cartão de crédito ou cheque
  - Dividir o caso de uso em seções
  - Para cada caso de uso:
    - Parte 1 – Resumo
    - Seção Principal
      - parte 2 – sequência típica de eventos
      - parte 3 – sequências alternativas
    - Seção Pagamento com dinheiro
      - parte 2 – sequência típica de eventos
      - parte 3 – sequências alternativas
    - Seção Pagamento com cartão de crédito
      - ...

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia


90

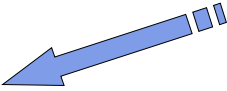
90

### Caso de Uso Comprar Itens

#### Seção Principal

#### Sequência típica de eventos



Ação do ator	Resposta do Sistema
1. Este caso de uso começa quando o Cliente chega ao TPV com itens para comprar	
2. O Caixa registra o identificador de cada item ...	3. Determina o preço do item...
4. Quando termina a entrada dos itens...	5. Calcula...
6. O Caixa informa o total ao cliente	
7. O Cliente escolhe o tipo de pagamento: i. Se for pagamento em dinheiro, ver seção <i>Pagamento em Dinheiro</i> ii. Se for pagamento com cartão de crédito ver seção <i>Pagamento por Cartão de Crédito</i> iii. Se for pagamento por cheque, ver seção <i>Pagamento em Cheque</i>	
	8. Registra a venda completada
9. O Caixa entrega o recibo para o Cliente	
10. O Cliente sai da loja com os itens...	

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia


91

91

### Caso de Uso Comprar Itens

#### Seção Pagamento com Dinheiro

#### Sequência típica de eventos



Ação do ator	Resposta do Sistema
1. O Cliente entrega o pagamento em dinheiro, possivelmente maior que o total da venda	
2. O Caixa registra a quantidade de dinheiro recebida	3. Exibe o valor do troco a ser devolvido ao cliente
4. O Caixa deposita o dinheiro recebido e retira o troco devido	
O Caixa entrega o troco ao Cliente	

**Sequência alternativa:**  
  
• **Linha 4:** Dinheiro insuficiente na gaveta para pagar o troco. Solicita dinheiro ao supervisor

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

92

92

## Planejar e Elaborar: Passos do Processo



1. Listar todas as funções (requisitos), definir os limites do sistema e identificar atores e casos de uso.
2. Escrever todos os casos de uso no formato de alto nível, classificando-os como principais, secundários e opcionais.
3. Desenhar o diagrama de casos de uso.
4. Escrever o formato expandido dos casos de uso mais importantes, mais complexos ou mais arriscados. Os demais poderão ser expandidos quando forem tratados em fases posteriores do processo de desenvolvimento.
5. Idealmente, postergar os casos de uso reais até a fase de projeto. Exceções podem ocorrer se:
  - a) descrições concretas auxiliam grandemente a compreensão, ou
  - b) os clientes demandam que o processo seja especificado dessa forma.

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

93

93

## Exemplo – Sistema TPV



(Passo 1. Identificar atores, casos de uso e limites do sistema.)

Os limites do sistema serão definidos como o sistema de hardware e software.

Atores e casos de uso:

**Caixa:** Abrir (Log In), Retirar dinheiro da caixa, Fechar

**Cliente:** Comprar Itens, Reembolsar Itens

**Gerente:** Iniciar e Encerrar (o sistema)

**Administrador do Sistema:** Adicionar novo usuário

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

94

94

## Exemplo – Sistema TPV



(Passo 2. Escrever casos de uso no formato de alto nível.)

Caso de uso: **Comprar Itens**

Atores: Cliente (iniciador), Caixa

Tipo: primário

Descrição: Um cliente chega ao balcão de saída da loja com itens para comprar. O caixa registra os itens de compra e recebe o pagamento. Quando termina, o cliente sai com os itens comprados.

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

95

95

## Exemplo – Sistema TPV



(Passo 2. Escrever casos de uso no formato de alto nível.)

Caso de uso: **Iniciar**

Atores: Gerente

Tipo: primário

Descrição: Um Gerente liga o sistema TPV de modo a prepará-lo para o uso pelos Caixas. O Gerente confere que as datas e hora estão corretas, após o que o sistema está pronto para uso dos Caixas.

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

96

96

## Exemplo: Sistema TPV

(Passo 3. Desenhar um diagrama de casos de uso.)

The diagram shows a central system boundary labeled 'TPV' containing several use cases: 'Comprar Itens', 'Abrir', 'Reembolsar Itens', 'Adicionar novos usuários', 'Iniciar', and 'etc.'. Four actors are connected to these use cases: 'Caixa' is connected to 'Comprar Itens', 'Abrir', and 'Reembolsar Itens'; 'Cliente' is connected to 'Comprar Itens' and 'Reembolsar Itens'; 'Administrador Do Sistema' is connected to 'Adicionar novos usuários'; and 'Gerente' is connected to 'Iniciar'.

unesp  
BCC  
2025

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

97

97

## Exemplo – Sistema TPV

unesp  
BCC  
2025

**Caso de Uso: Comprar Itens**

**Atores:** Cliente (iniciador), Caixa

**Propósito:** Captura a venda e seu pagamento em dinheiro

**Visão geral:** Um cliente chega a um ponto de pagamento, com vários itens que deseja comprar. O caixa registra os itens de compra e recebe o pagamento, o qual pode necessitar autorização. No final, o cliente sai com os itens comprados.

**Tipo:** primário e essencial

**Referências**

**Cruzadas:** Funções: R1.1, R1.2, R1.3, R1.7, R1.9, R2.1, R2.2, R2.3, R2.4

**Casos de Uso:** o caixa deve ter completado o caso de uso Abrir

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

98


98

## Exemplo – Sistema TPV

Seção : Principal

Sequência Típica de Eventos

Ação do ator	Resposta do sistema
1. Este caso de uso começa quando um Cliente chega a um ponto de pagamento equipado com um TPV, com vários itens que deseja comprar.	
2. O Caixa registra cada item.	3. Determina o preço do item e acrescenta informação sobre o item à transação de vendas em andamento
Se houver mais de um exemplar do item, o Caixa também pode entrar a quantidade.	A descrição e o preço do item corrente são apresentados
4. No término da entrada de itens, o Caixa indica para o TPV que a entrada de itens está completa.	5. Calcula e apresenta o total da venda.
6. O Caixa informa ao Cliente o total.	



Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

99

## Exemplo – Sistema TPV


(continuação...)

Ação do ator	Resposta do sistema
7. O Cliente escolhe o tipo de pagamento: a. Se pagamento em dinheiro, ver seção <i>Pagar com Dinheiro</i> . b. Se pagamento com cartão, ver seção <i>Pagar com Cartão de Crédito</i> . c. Se pagamento com cheque, ver seção <i>Pagar com Cheque</i> .	
11. O Caixa dá o recibo ao Cliente.	8. Registra a venda completada.
12. O Cliente sai com os itens comprados.	9. Atualiza os níveis de estoque.
	10. Gera um recibo.

**Sequências alternativas**

**Linha 2:** Entrada de Identificador de item inválido. Indica erro.

**Linha 7:** Cliente não pode pagar. Cancelar a transação de venda.




Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

100

## Exemplo – Sistema TPV

Seção: Pagar com Dinheiro

Sequência Típica de Eventos



**Ação do ator**

1. O Cliente dá um pagamento em dinheiro.  
O valor fornecido é possivelmente maior que o total da venda.

2. O Caixa registra a quantia fornecida

4. O Caixa deposita o dinheiro recebido e retira o troco devido.  
O Caixa dá o troco ao Cliente

*Sequências Alternativas*

**Linha 1:** O Cliente não tem dinheiro suficiente. Pode cancelar a venda ou iniciar outro método de pagamento

**Linha 4:** A gaveta de dinheiro não contém o suficiente para pagar o troco. O Caixa solicita mais dinheiro ao supervisor ou pede ao Cliente uma quantia de dinheiro diferente ou a opção por um outro método de pagamento

**Resposta do sistema**

3. Apresenta o troco devido ao Cliente

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia


101

101

## Exemplo – Sistema TPV

Seção: Pagar com Cartão de Crédito

Sequência Típica de Eventos



**Ação do ator**

1. O Cliente comunica suas informações de Crédito para o pagamento com cartão de crédito

3. O SAC autoriza o pagamento

**Resposta do sistema**

2. Gera uma solicitação de pagamento com cartão de crédito e a envia a um Serviço de Autorização de Crédito (SAC) externo

4. Recebe uma resposta de aprovação de crédito do SAC.

5. Lança o pagamento com cartão de crédito e a informação da resposta de aprovação no sistema de Contas a Receber (C/R). (O SAC deve dinheiro à Loja, logo C/R deve fazer acompanhamento)

6. Exibe a mensagem de autorização bem sucedida

*Sequências Alternativas*

**Linha 3:** Solicitação de crédito negada pelo SAC. Sugerir um método de pagamento diferente

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

102

102



# Exemplo – Sistema TPV


## Seção: Pagar com Cheque

*Sequência Típica de Eventos*

Ação do ator	Resposta do sistema
1. O Cliente preenche um cheque e se identifica.	
2. O Caixa registra a informação de identificação e solicita autorização para pagamento com cheque	3. Gera uma solicitação de pagamento com cheque e a envia a um Serviço de Autorização de Cheques externo
4. O Serviço de autorização de Cheques autoriza o pagamento	5. Recebe uma resposta de aprovação do Serviço de Autorização de Cheques.
	6. Indica autorização bem-sucedida.

**Sequências Alternativas**

**Linha 4:** Solicitação de cheque negada pelo Serviço de Autorização de Cheques. Sugerir um método de pagamento diferente




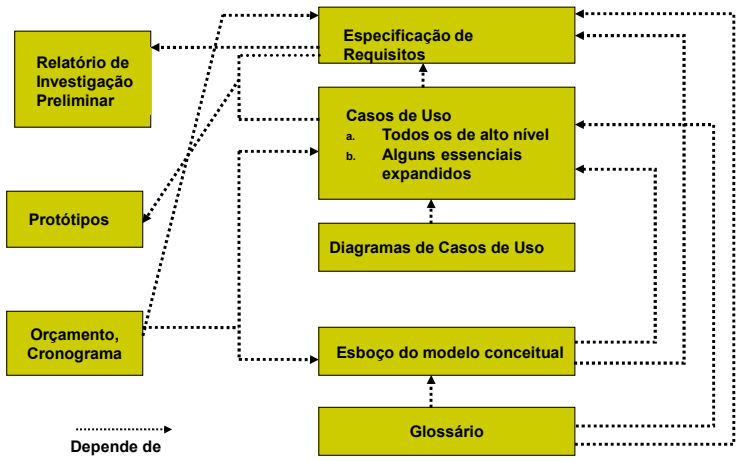
Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

103

103

# Planejar e Elaborar: Artefatos





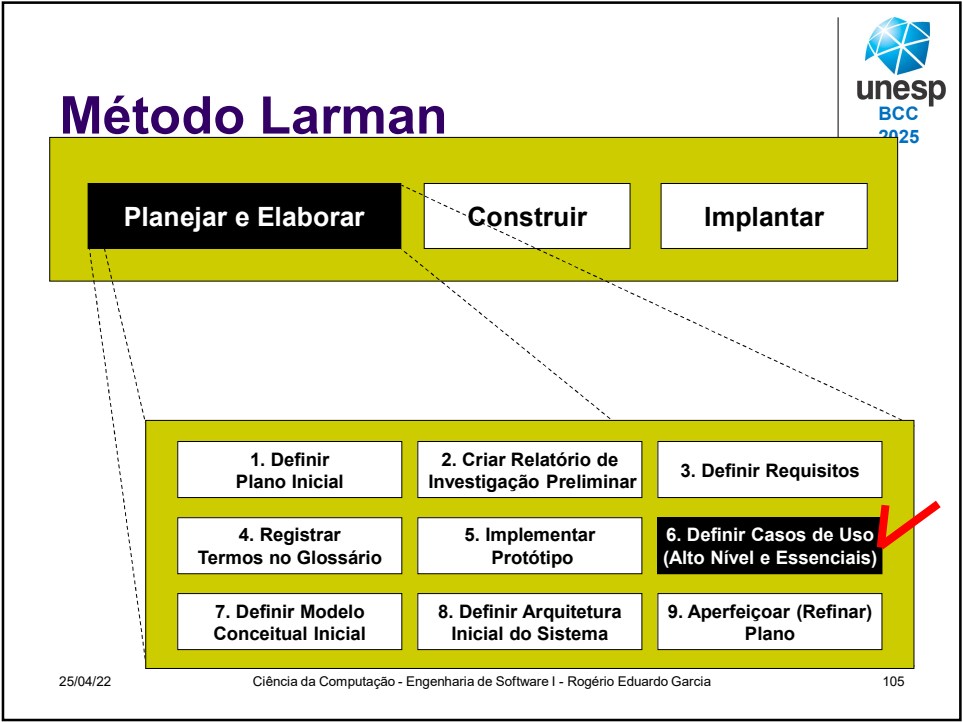
Depende de

25/04/22

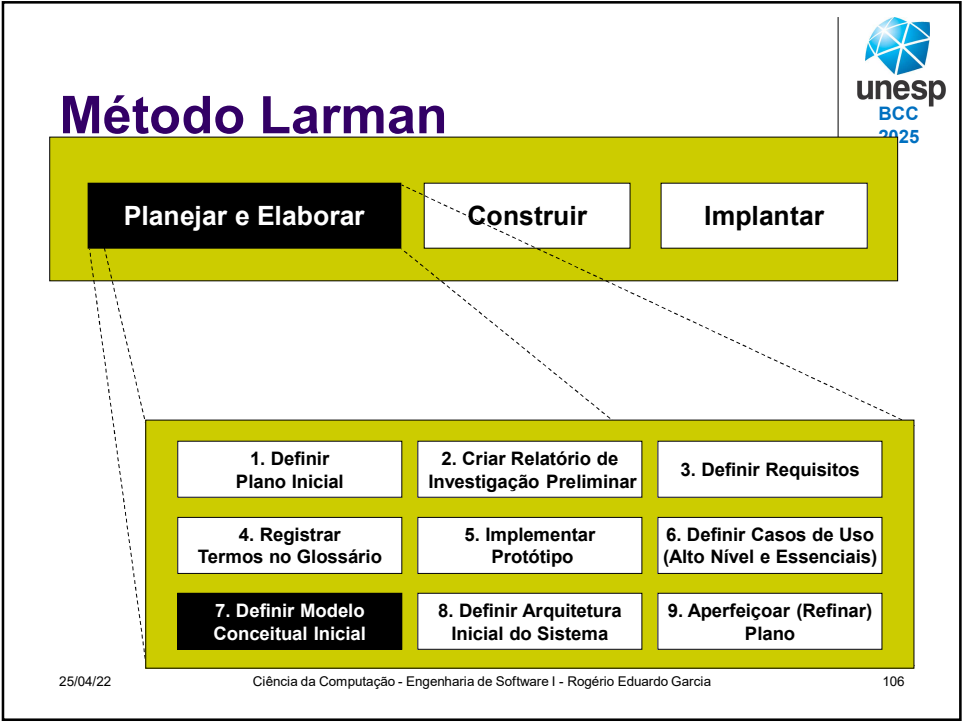
Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

104

104



105



106

## Modelos Conceituais



- **Modelo Conceitual** é uma representação dos conceitos, ou objetos, do mundo real pertencentes a um domínio de interesse
- É exibido por um conjunto de diagramas de estrutura **estática**, no qual não se definem operações
- Pode ser tratado como um “dicionário visual” das abstrações significativas do domínio
  - ajuda a compreender vocabulário e informação do domínio
- Pode mostrar: conceitos, associações entre conceitos e atributos de conceitos

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

107

107

## Modelo Conceitual: Conceito



- Informal: idéia, “coisa”, ou objeto do mundo real no domínio de interesse.
  - Algo digno de nota, de ser documentado, de importância para o domínio.
- Formal: Um conceito pode ser considerado em termos de seu:
  - Símbolo: palavra ou imagem representando um conceito.
    - Ex.: Venda
  - Intenção: a definição de um conceito.
    - Ex.: Uma venda representa uma transação de compra e possui data e hora.
  - Extensão: o conjunto de exemplos (instâncias) ao qual o conceito se aplica.
    - Ex.: Venda1, Venda2, Venda3 ...


25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

114

114

## Conceitos no Sistema TPV




TPV      Venda      Loja

- Como identificar conceitos em um sistema ?

25/04/22      Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia      116

116

## Estratégias para Identificar Conceitos



- É melhor especificar em excesso um modelo conceitual com muitos conceitos do que subespecificá-lo.
  - Menos conceitos não implicam em um modelo melhor.
  - Não exclua um conceito só porque sua necessidade não está óbvia nos requisitos.
  - Não exclua um conceito só porque não tem atributos – ele pode possuir um papel de comportamento e não de informação.
- Usar uma Lista de **Categorias de Conceitos**.
- Identificar **Substantivos**.

25/04/22      Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia      117

117

## Conceito - Definição



- Informal: idéia, “coisa”, ou objeto do mundo real no domínio de interesse
  - algo digno de nota, de ser documentado, de importância para o domínio
- Formal - Um conceito pode ser considerado em termos de seu:
  - **Símbolo**: palavra ou imagem representando um conceito. Ex: *Venda*
  - **Intenção**: a definição de um conceito. Ex: *Uma venda representa uma transação de compra e possui data e hora*
  - **Extensão**: o conjunto de exemplos (instâncias) ao qual o conceito se aplica: Ex: *Venda1, Venda2, Venda3 ...*

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

118

118

## Exemplo



- **Símbolo** - Ex: *Aeronave*
- **Intenção** - Ex: (o conceito) *Aeronave representa uma aeronave, ou seja, um meio de transporte aéreo que possui categoria, dimensões, número de lugares, ...*
- **Extensão** - Ex: *AirBus PT999, Boeing747 PX111, ...*


25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

119

119

## Conceitos no Sistema TPV



- Como identificar conceitos em um sistema ?

25/04/22      Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia      120

120

## Exemplos de Categorias de Conceitos

- Objetos físicos ou tangíveis: TPV, Carro, Aeronave
- Especificações ou Descrições de “Coisas”: EspecificaçãoProduto, ListaVerificação
- Lugares: Loja, Aeroporto
- Transações: Venda, Pagamento, Reserva
- Regras e Políticas: PolíticaReembolso
- Itens de linha de transação: ItemLinhaVendas

25/04/22      Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia      121

121

## Exemplos de Categorias de Conceitos (cont.)



- Papéis desempenhados por pessoas: Caixa
- Contêineres: Depósito, Armário, Aeronave
- Coisas em um contêiner: Item, Passageiro
- Catálogos: CatálogoProdutos, CatálogoPeças
- Organizações: DepartamentoVendas
- Sistema externo: SismAutorizaçãoCartCrédito
- ...

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

122

122

## Identificação de Substantivos - caso de uso *Comprar Itens*



1. Este caso de uso começa quando um **Cliente** chega a um **ponto de pagamento equipado com um TPV** com vários **itens** que deseja comprar.
2. O **caixa** registra o **código universal do produto** (UPC) de cada **item**.  
Se houver mais de um exemplar do **item** o **caixa** também pode entrar a **quantidade**.
3. Determina o preço do **item** e acrescenta informação sobre o **item** à **transação de vendas** em andamento.  
A **descrição** e o **preço do item** corrente são apresentados

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

123

123

## Identificação de Substantivos



Lembre-se:

1. Nem todos os substantivos são conceitos – linguagem natural pode ser ambígua  
Ex: substantivos diferentes podem representar o mesmo conceito – (*Consumidor e Cliente*)
2. Alguns dos substantivos são candidatos a conceitos e outros são candidatos a atributos

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

124

124

## Conceitos candidatos

### Domínio TPV – caso de uso *Comprar Itens*



⇒ Ideal: Combinar as estratégias para identificar uma lista de candidatos a conceito

- TPV
- Item
- Loja
- Venda
- CatálogoProdutos
- EspecificaçãoProduto
- ItemLinhaVenda
- Caixa
- Cliente
- Pagamento
- Gerente

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

125

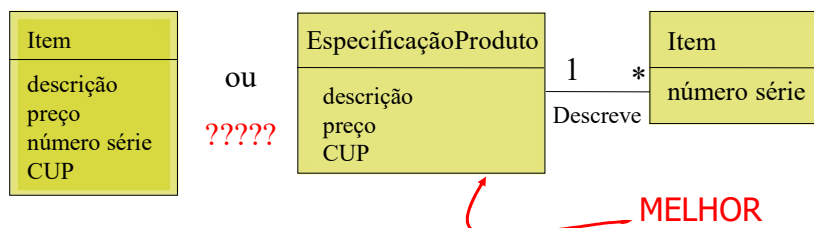
125



## O Conceito de Especificação (ou de Descrição)



- O que aconteceria se todas as TVs de 19" (Item) fossem vendidas ?
- Como fazer para saber o preço desse item de venda ?



25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

127

127

## Quando utilizar conceitos de especificações



- Quando houver necessidade da existência de uma descrição de um item ou serviço, independente da existência de uma instância do item ou serviço
  - se a exclusão de instâncias do item/serviço resultar em perda de informação que deveria ser mantida
- Quando reduzir informação redundante ou duplicada

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

128

128

unesp  
BCC  
2025

## Exemplo para TPV

- Parte do Modelo Conceitual do TPV
  - indica que os conceitos *Venda* e *Pagamento* são significativos para o domínio
  - indica que *Venda* e *Pagamento* estão relacionados de uma forma digna de nota

Venda

data

hora

1

Pago-por

1

Pagamento

quantia

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

131

131

unesp  
BCC  
2025

## Modelo Conceitual

- Modelo Conceitual NÃO mostra artefatos de software
  - objetos ou classes de software
  - métodos ou responsabilidades
  - janelas
  - bases de dados

NÃO!!!

BDdeVendas

Venda

data

hora

imprimir ()

25/04/22


Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

132

132

## Cardinalidade ou Multiplicidade

- Cardinalidade define quantos objetos participam da relação
  - É o número de instâncias de objetos da classe que participam da relação
  - Para cada associação e agregação, são definidas duas multiplicidades: uma para cada participante do relacionamento.



25/04/22


Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

133

133

## Multiplicidade: Exemplo

- A multiplicidade define quantas instâncias de um conceito A podem ser associadas a cada instância do conceito B



*	C	zero ou mais - muitos (as)
1..*	C	um ou mais
1..40	C	um a quarenta
5	C	exatamente cinco
3,5,8	C	exatamente três, cinco ou oito

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

134

134

## Navegação



- Embora associações e agregações sejam bi-direcionais por *default*, por ser desejável restringir a navegação em uma direção
  - Para isso, uma ponta (flecha) é adicionada à linha, indicando a direção da navegação

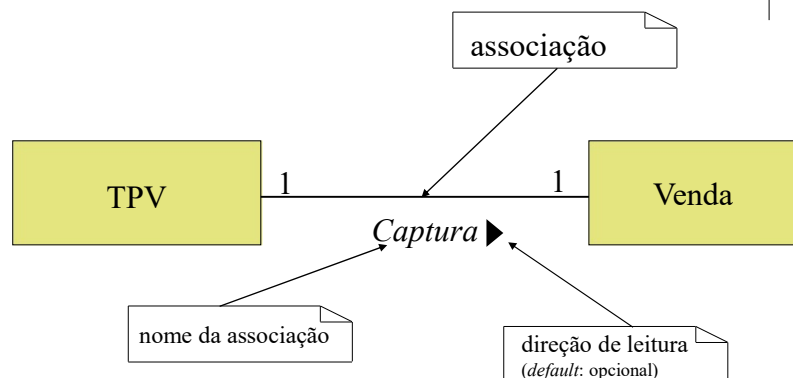
25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

135

135

## Exemplo do Sistema TPV



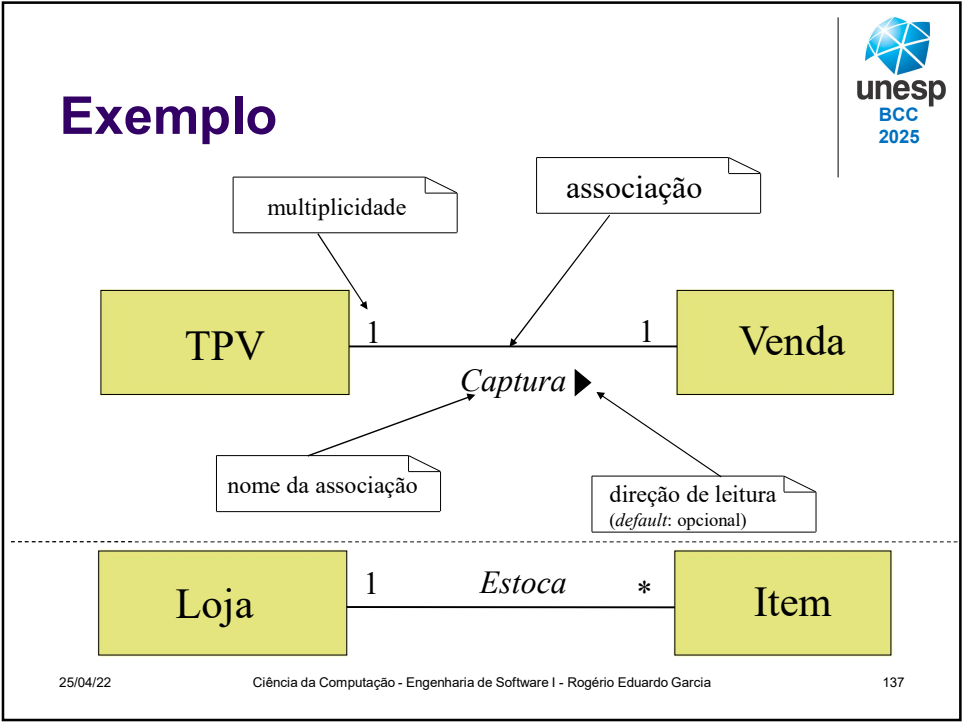
OBS: o símbolo  SOMENTE indica direção de leitura – não tem significado no modelo

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

136

136



137

### Associação - definição

- **Associação** é um relacionamento entre conceitos
  - indica uma conexão com significado e interesse
- Em UML são descritas como “relacionamentos semânticos entre objetos diferentes”

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

138

138

## Critérios para incluir associações



- Quando o conhecimento associado necessita ser preservado por algum tempo
  - “necessário-ser-conhecida” – requisitos indicam essa necessidade
  - Ex: associação entre Venda e Pagamento
- Evite associações cuja necessidade não é sugerida nos requisitos
  - Ex: associação entre Venda e Gerente
- É mais importante identificar conceitos do que associações
- Excesso de associações pode tornar o modelo conceitual confuso
- Evite mostrar associações redundantes ou deriváveis

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

139

139

## Associações Comuns



- A é uma parte física de B
  - Gaveta – TPV
  - Asa - Aeronave
- A é uma parte lógica de B
  - ItemLinhaVenda – Venda
  - PernaVôo (Flight Leg) - RotaVôo
- A está fisicamente contida em/sobre B
  - Item – Prateleira
  - Passageiro - Aeronave
- A está logicamente contida em B
  - DescriçãoItem - Catálogo
  - Vôo – ProgramaçãoVôo
- A é registrada em B
  - Venda - TPV
  - Reserva - ManifestoVôo

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

140

140

## Associações Comuns



- A é uma descrição para B
  - DescriçãoItem – Item
  - DescriçãoVoo – Voo
- A é um item de linha de uma transação ou relatório B
  - ItemLinhaVenda – Venda
  - ServiçoManutenção - LogManutenção
- A é uma transação relacionada a outra transação B
  - Pagamento – Venda
  - Reserva – Cancelamento
- ...

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

141

141

## Associações com Papéis



- Cada extremo de uma associação é chamado de **papel**.
- Os papéis podem ter, opcionalmente, as seguintes propriedades:
  - Nome
  - Expressão de multiplicidade
  - Navegabilidade

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

142

142

unesp  
BCC  
2025

## Associações com Papéis

- Nomes de papéis são necessários, principalmente, para associação entre dois objetos de **mesma classe**.

```
classDiagram
    class Companhia
    class Empregado
    Companhia "1" -- "1..*" Empregado : subordinado
    Empregado "0..*" -- "1" Empregado : 1
```

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

143

143

unesp  
BCC  
2025

## Associações Múltiplas entre Conceitos

```
classDiagram
    class Vôo
    class Aeroporto
    Vôo "*" -- "1" Aeroporto : Voa-para
    Vôo "*" -- "1" Aeroporto : Voa-de
```

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

144

144



unesp  
BCC  
2025

## Associações e Implementação

- Uma associação indica um relacionamento significativo apenas sob a **perspectiva conceitual**.
  - Uma associação não implica em um fluxo de dados ou conexão entre objetos em uma solução de software.
  - Algumas associações do modelo conceitual podem não ser necessárias na implementação.
  - Durante a implementação podem ser descobertas associações entre objetos de software que foram esquecidas durante a modelagem conceitual.

25/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

145

145

unesp  
BCC  
2025

## Exemplo

```
classDiagram
    class CompanhiaAerea[Companhia Aérea]
    class Pessoa
    class Voo
    class Aeronave
    CompanhiaAerea "1" -- "1..*" Pessoa : Emprega
    Pessoa "1" -- "*" Pessoa : supervisiona
    Pessoa "1" -- "*" Voo : Designada-para
    Voo "*" -- "1" Aeronave : Designada-para
```

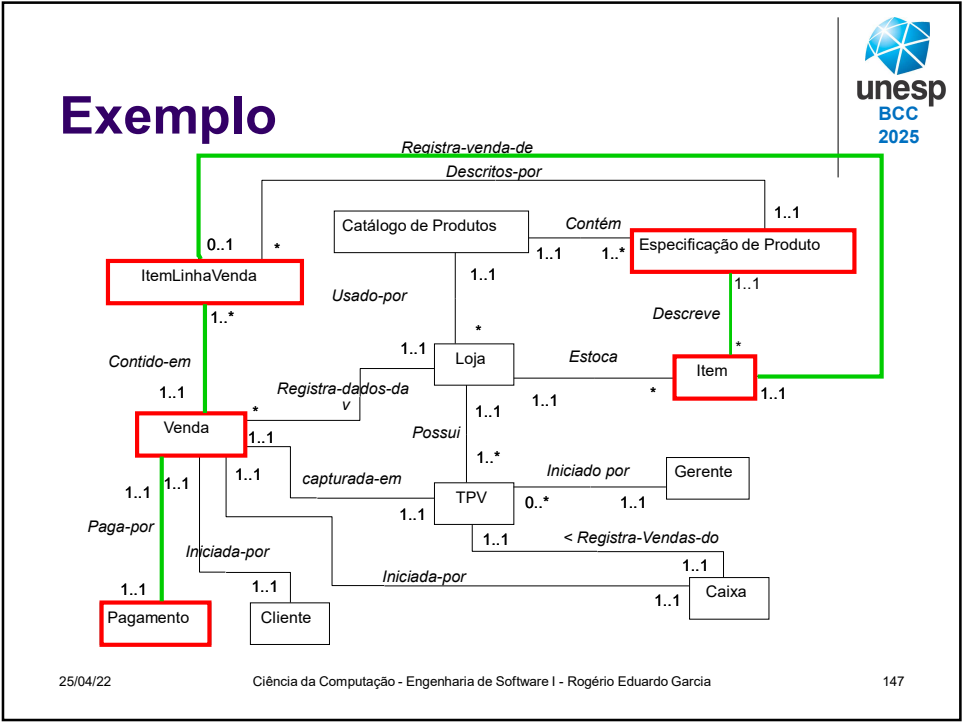
OBS: considere vôos não simultâneos na alocação de pessoas e aeronaves. Essa alocação seria feita, por exemplo, o planejamento de vôos do dia/semana/mês...

25/04/22

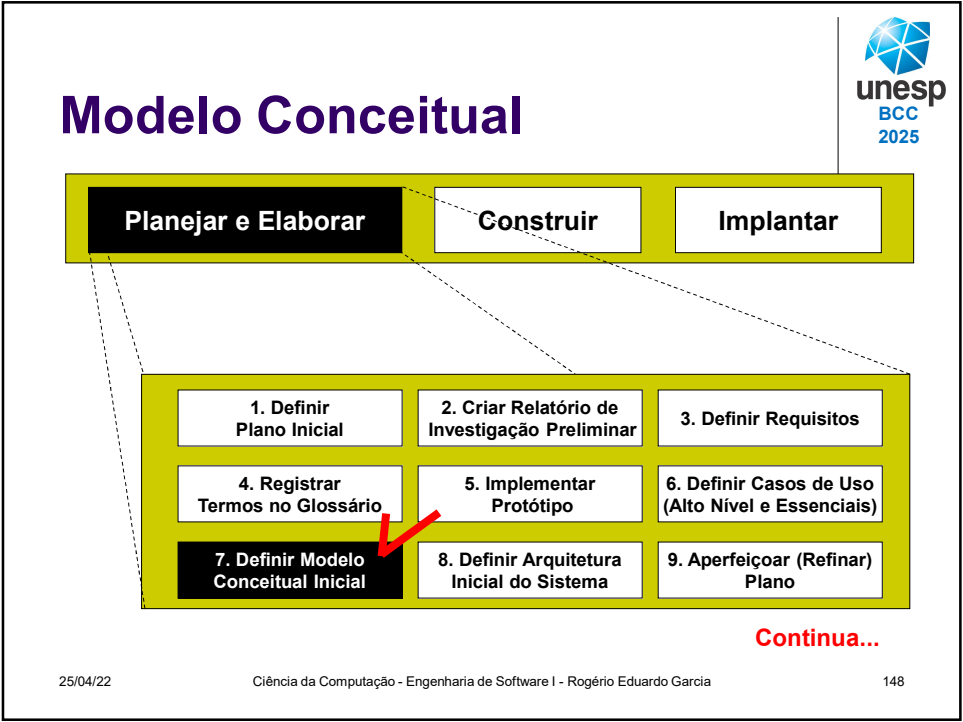
Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

146

146



147



148