

# ENGENHARIA DE SOFTWARE 2 – AULA 4

---

PROF<sup>a</sup> M<sup>a</sup> DENILCE VELOSO  
DENILCE.VELOSO@FATEC.SP.GOV.BR  
DENILCE@GMAIL.COM

# ENGENHARIA DE SOFTWARE – ATIVIDADES DO PROCESSO DE SOFTWARE

---

Apesar da diversidade, todo Processo de Software **devem incluir de alguma forma as quatro atividades fundamentais:**

1. Especificação (por exemplo Elicitação pode estar incluída como subatividade aqui)
2. Desenvolvimento
3. Validação
4. Evolução

\*\* incluem subatividades

Fonte: (Sommerville, 2019 p.30)



# ENGENHARIA DE REQUISITOS – ELICITAÇÃO DE REQUISITOS

---



# ENGENHARIA DE REQUISITOS - ELICITAÇÃO DE REQUISITOS

- 
- Após um estudo inicial de viabilidade, o próximo estágio da engenharia de requisitos é a elicitação dos requisitos.
  - Às vezes chamada de elicitação de requisitos ou descoberta de requisitos.
  - Envolve a equipe técnica trabalhando com os clientes para encontrar o domínio da aplicação, os serviços que o sistema deve fornecer e as restrições operacionais do sistema.
  - Pode envolver usuários finais, gerentes, engenheiros envolvidos em manutenção, especialistas de domínio, sindicatos, etc. **Estes são chamados de stakeholders.**



# ELICITAÇÃO DE REQUISITOS - PROBLEMAS

---

- *Stakeholders* **não sabem** o que realmente querem.
- *Stakeholders* expressam os requisitos da **sua maneira**.
- *Stakeholders* podem ter **conflitos** de requisitos entre eles.
- Fatores **organizacionais** e **políticos** podem influenciar os requisitos de sistema.
- Os requisitos **mudam** durante o processo de análise.
- **Novos stakeholders** podem surgir e o ambiente de negócio pode mudar.



# O PROCESSO DE ELICITAÇÃO E ANÁLISE DE REQUISITOS

AS ETAPAS INCLUEM:

---



Fonte: (Sommerville, 2019 p.97)



# ELICITAÇÃO DE REQUISITOS - PROBLEMAS

---

- Descoberta de requisitos

**Interagir com os *stakeholders*** para descobrir seus requisitos.

Os **requisitos de domínio**<sup>1</sup> também são descobertos neste estágio.

- Classificação e organização de requisitos

**Agrupar os requisitos relacionados** e os organiza em grupos coerentes.

- Priorização e negociação

**Priorização de requisitos e resolução de conflitos** de requisitos.

- Especificação de requisitos

Os **requisitos são documentados** e inseridos na próxima rodada de a espiral. (iterativo)

1 - Exemplo: Um aluno pode se matricular em uma disciplina desde que ele tenha sido aprovado nas disciplinas consideradas pré-requisitos.

# ELICITAÇÃO DE REQUISITOS - TÉCNICAS

---

- Entrevistas
- Etnografia
- Histórias e cenários



# ELICITAÇÃO DE REQUISITOS - ENTREVISTAS

- Entrevistas **formais** ou **informais** com os *stakeholders* fazem parte da maioria dos processos de Engenharia de Requisitos.
- 

- Tipos de entrevistas
  - **Fechadas com base em uma lista pré-determinada de perguntas.**
  - **Abertas** onde várias questões são exploradas com *stakeholders*.
- Entrevista eficaz
  - **Ter a mente aberta**, evitando ideias pré-concebidas sobre o requisitos e estar disposto a ouvir os *stakeholders*.
  - Pedir ao entrevistado para iniciar as discussões **usando uma questão inicial**, uma proposta de requisitos ou trabalhando juntos em um sistema de protótipo.



# ELICITAÇÃO DE REQUISITOS – HISTÓRIAS E CENÁRIOS

- ~~Cenários e histórias (essencialmente a mesma coisa)~~, são **exemplos reais** de como um sistema é utilizado (sequência de eventos).
- São uma **descrição de como um sistema pode ser utilizado** em uma tarefa específica.
- Por serem baseados em uma situação prática, os *stakeholders* podem se relacionar com eles e podem comentar sobre sua situação (participação) em relação à história.
- Parte das histórias podem ser representadas como cenários.
- **As histórias não se preocupam com a linguagem técnica.**



**DV0**

Histórias - narrativas sem se preocupar muito com a linguagem técnica. Cenários - interações entre o usuário e o sistema e são escritos em uma linguagem mais técnica.

DENILCE VELOSO; 2023-03-16T11:55:18.586

# ELICITAÇÃO DE REQUISITOS - ETNOGRAFIA

---

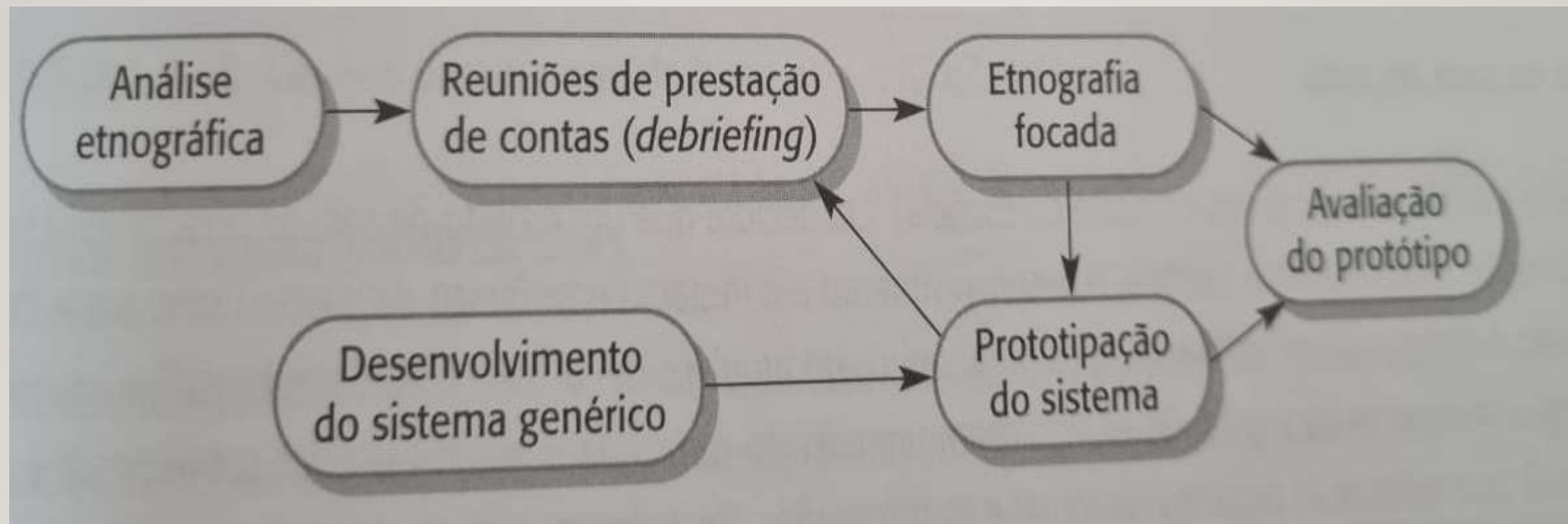
- Observação ou etnografia, em que se **observa as pessoas executando seu trabalho** para ver quais artefatos elas usam, como usam, etc.
- Importante para compreender os sistemas existentes, mas essa compreensão nem sempre ajuda na inovação.



# ELICITAÇÃO DE REQUISITOS - ETNOGRAFIA

---

A etnografia pode ser combinada com o desenvolvimento de um protótipo do sistema.



**DV0**

etnografia é uma abordagem de pesquisa que busca entender a cultura, comportamentos e interações humanas em um determinado contexto, etnografia focada é a concentração em um tema

DENILCE VELOSO; 2023-03-16T11:58:38.816



# ENGENHARIA DE REQUISITOS – ESPECIFICAÇÃO DE REQUISITOS

---



# ESPECIFICAÇÃO DE REQUISITOS

- 
- É o processo que deve escrever os requisitos de usuário e de sistema em um documento de requisitos.
  - Os requisitos do usuário devem ser compreensíveis pelos usuários finais e clientes que não possuem conhecimento técnico.
  - Os requisitos do sistema são requisitos mais detalhados e pode incluir mais informações técnicas.
  - Os requisitos podem fazer parte de um contrato para o desenvolvimento de sistema.
  - Portanto, é importante que estes sejam tão completos quanto possível.



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS

Notação	Descrição
Sentenças em linguagem natural	Os requisitos são escritos usando frases numeradas em linguagem natural. Cada frase deve expressar um requisito.
Linguagem natural estruturada	Os requisitos são escritos em linguagem natural em um formulário ou <i>template</i> . Cada campo fornece informações sobre um aspecto do requisito.
Notações gráficas	Modelos gráficos, suplementados por anotações em texto, são utilizados para definir os requisitos funcionais do sistema. São utilizados com frequência os diagramas de casos de uso e de sequência da UML.
Especificações matemáticas	Essas notações se baseiam em conceitos matemáticos como as máquinas de estados finitos ou conjuntos. Embora essas especificações inequívocas possam reduzir a ambiguidade em um documento de requisitos, a maioria dos clientes não compreende uma especificação formal. Eles não conseguem averiguar se ela representa o que desejam e relutam em aceitar essa especificação como um contrato do sistema (discutirei essa abordagem no Capítulo 10, que aborda a dependabilidade do sistema).

# NOTAÇÕES PARA **ESPECIFICAÇÃO** DE REQUISITOS - EXEMPLOS

Linguagem Natural:

"O sistema deve permitir que o usuário cadastre um novo cliente, fornecendo seu nome, endereço e número de telefone."

Linguagem Natural Estruturada:

"Como vendedor, eu quero ser capaz de adicionar um novo cliente ao sistema, preenchendo um formulário que inclui campos para o nome, endereço e número de telefone, de modo que eu possa manter um registro preciso dos clientes."

Notação gráfica - caso de uso (diagrama de caso de uso), descrição de caso de uso.

Matemática - quando envolve fórmula - O software deve ser capaz de calcular o lucro semanal, seguindo a fórmula tal, onde  $x$  significa a produção das máquinas.



**DV0**

Linguagem Natural:

"O sistema deve permitir que o usuário cadastre um novo cliente, fornecendo seu nome, endereço e número de telefone."

Linguagem Natural Estruturada:

"Como vendedor, eu quero ser capaz de adicionar um novo cliente ao sistema, preenchendo um formulário que inclui campos para o nome, endereço e número de telefone, de modo que eu possa manter um registro preciso dos clientes."

Notação gráfica - caso de uso (diagrama de caso de uso), descrição de caso de uso

Matemática - quando envolve fórmula - O software deve ser capaz de calcular o lucro semanal, seguindo a fórmula tal, onde x significa a produção das máquinas.

DENILCE VELOSO; 2023-03-16T12:09:45.757



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – LINGUAGEM NATURAL

---

- Linguagem Natural, exemplo:

O sistema deve medir o nível de açúcar no sangue e fornecer insulina, se for necessário, a cada 10 minutos.

O sistema pode receber os arquivos nos formatos txt, doc e xml.

*\*\* usar “deve” para requisitos obrigatórios e “pode” para requisitos desejáveis*



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – LINGUAGEM NATURAL ESTRUTURADA

- Linguagem Natural Estruturada, exemplo:

Bomba de insulina/Software de controle/SRS/3.3.2	
Função	Computar a dose de insulina: nível de açúcar seguro.
Descrição	Computa a dose de insulina a ser fornecida quando o nível de açúcar atual estiver na zona segura entre 3 e 7 unidades.
Entradas	Leitura atual do açúcar (r2), as duas leituras prévias (r0 e r1).
Fonte	Leitura atual de açúcar do sensor. Outras leituras da memória.
Saídas	DoseComp – a dose de insulina a ser fornecida.
Destino	Laço de controle principal.
Ação	DoseComp é igual a zero se o nível de açúcar estiver estável ou caindo; ou se o nível estiver aumentando, mas a taxa de crescimento estiver diminuindo. Se o nível estiver aumentando e a taxa de crescimento também, então a DoseComp é obtida pela divisão por 4 da diferença entre o nível de açúcar atual e o nível anterior, arredondando o resultado. Se o resultado for arredondado para zero, então a DoseComp é definida como dose mínima que pode ser fornecida (ver Figura 4.14).
Requer	Duas leituras prévias para que a taxa de variação do nível de açúcar possa ser calculada.
Pré-condição	O reservatório de insulina contém pelo menos a dose máxima permitida.
Pós-condição	r0 é substituída por r1, então r1 é substituída por r2.
Efeitos colaterais	Nenhum.

# PROBLEMAS COM A LINGUAGEM NATURAL

---

- Falta de Clareza

Difícil conseguir precisão sem tornar o documento de difícil leitura.

- Confusão de Requisitos

Requisitos funcionais e não funcionais tendem a estarem misturados.

- Fusão de requisitos

Vários requisitos diferentes podem ser expressos juntos como um único requisito.



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – NOTAÇÕES GRÁFICAS – CASOS DE USO

---

- Segundo Pressman (2011) na documentação da Especificação de Requisitos os Requisitos Funcionais e os Requisitos Não Funcionais são documentados; nesta etapa podem ser utilizados Diagramas de Casos de Uso (UML).
- Os casos de uso são uma maneira de escrever as interações entre os usuários e um sistema usando um modelo gráfico e um texto estruturado. (Sommerville, 2019).



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – NOTAÇÕES GRÁFICAS – CASOS DE USO

---

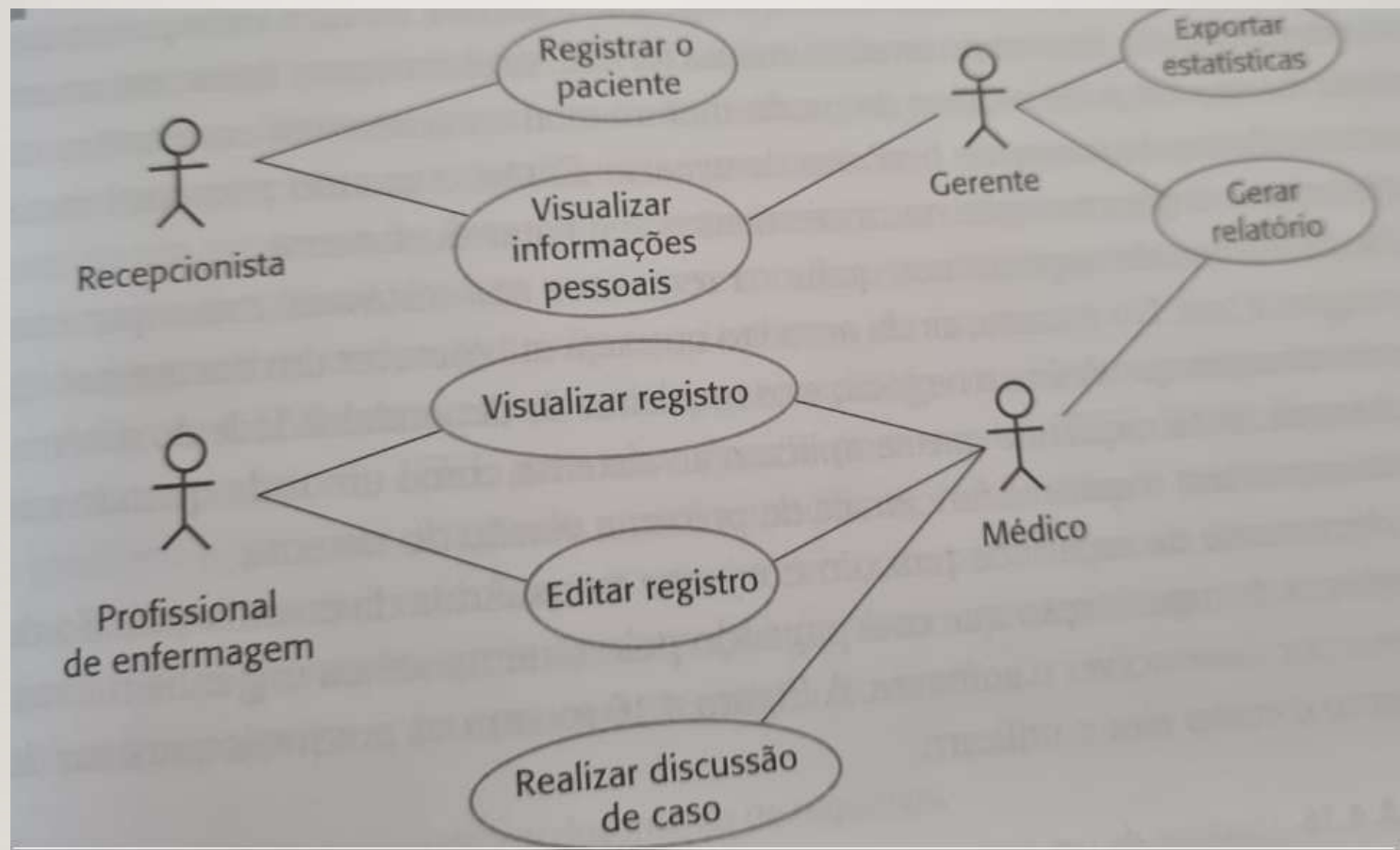
O caso de uso contém o nome da funcionalidade e pode conter sua respectiva descrição. No caso de uso será definido o ator de maneira que se relacione com as funcionalidades, a partir dessa fase ficam melhor descritos os requisitos, para então iniciar a modelagem do banco de dados, desenvolvimento e outras etapas.





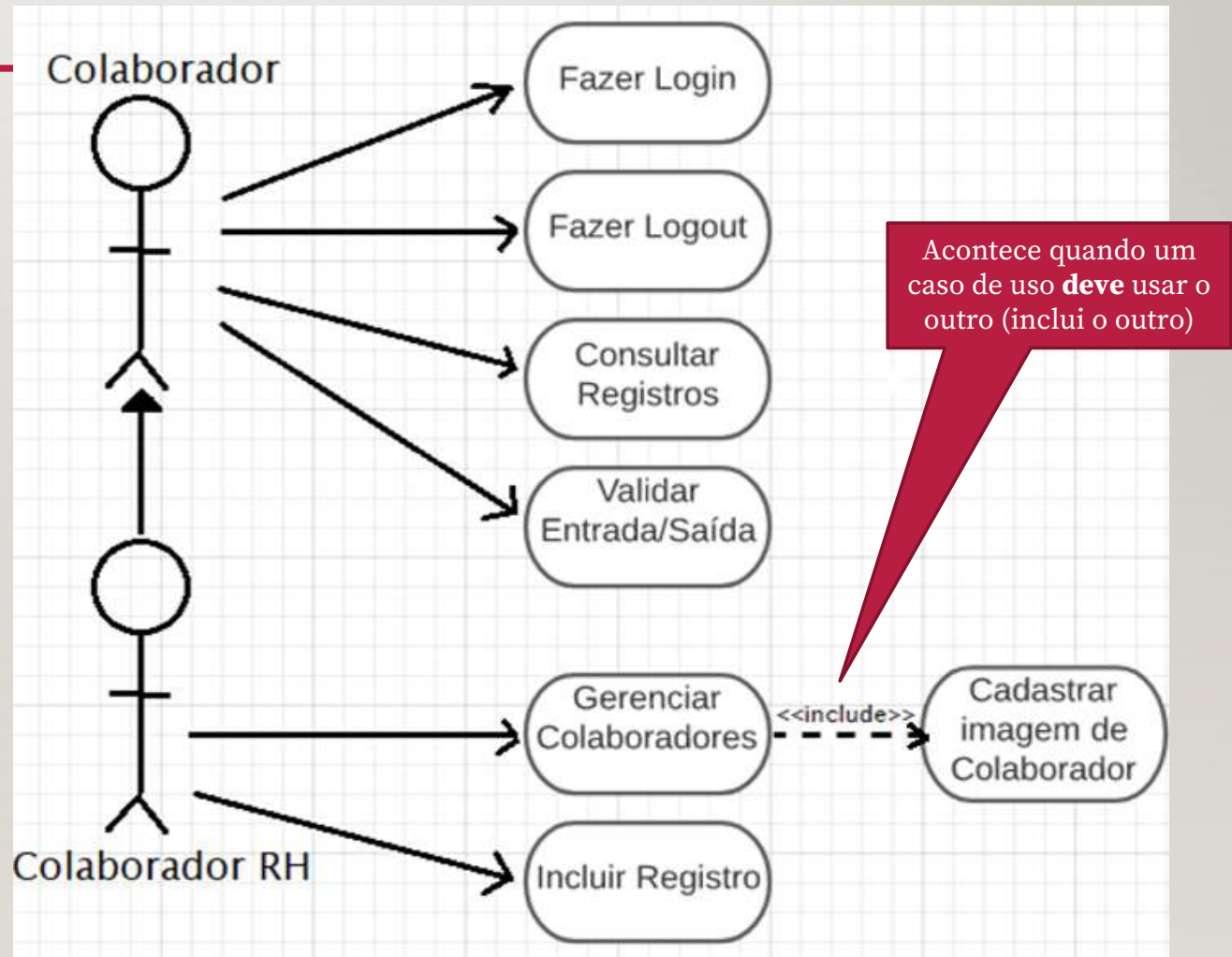
# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – NOTAÇÕES GRÁFICAS – CASOS DE USO

- Exemplo: Sistema MentCare



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – NOTAÇÕES GRÁFICAS – CASOS DE USO

- Exemplo: Sistema Controle de Ponto usando biometria

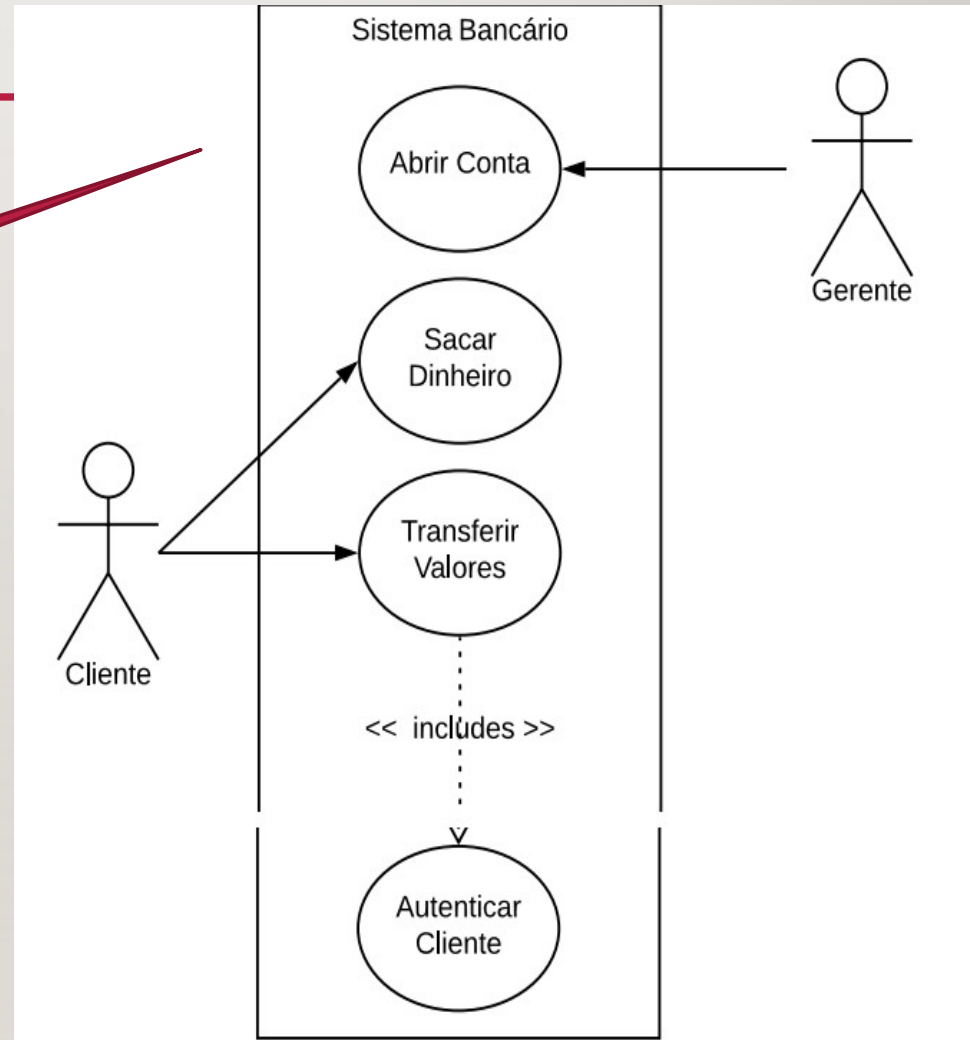




# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – NOTAÇÕES GRÁFICAS – CASOS DE USO

- Exemplo: Sistema Bancário

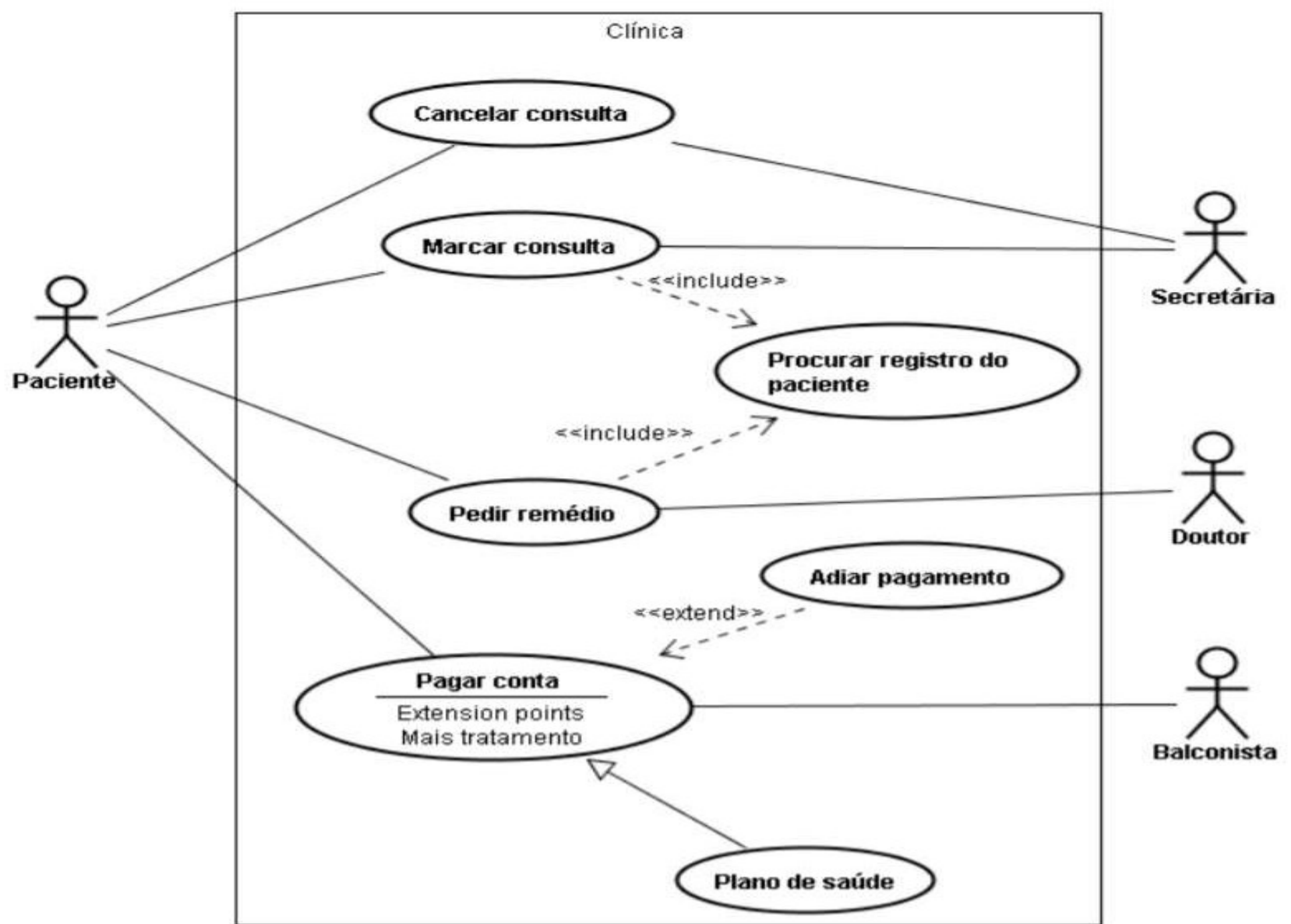
A fronteira do sistema é opcional



# NOTAÇÕES PARA ESPECIFICAÇÃO DE REQUISITOS – NOTAÇÕES GRÁFICAS – CASOS DE USO

---

- Exemplo: Sistema Clínica → próximo slide



# DOCUMENTO DE REQUISITOS DE SOFTWARE

- O Documento de Requisitos de Software é um compromisso. Ele tem o papel de comunicar os requisitos levantados, especificar para os desenvolvedores o que deve ser implementado e até mesmo prever a evolução do sistema.
- O PADRÃO IEEE/ANSI 830/1998 é uma estrutura genérica proposta pela IEEE, mas não é algo mandatório, cada organização adapta o formato do documento para suas necessidades.

\*\* IEEE – *Institute of Electrical and Electronic Engineers* colabora no incremento da prosperidade mundial, promovendo a engenharia de criação, desenvolvimento, integração, compartilhamento e o conhecimento aplicado no que se refere à ciência e tecnologias da eletricidade e da informação, em benefício da humanidade e da profissão.



# FORMATO SUGERIDO PELO PADRÃO IEEE/ANSI 830/1998.

## 1. Introdução

1. Propósito do documento de requisitos
2. Escopo do produto
3. Definições, siglas e abreviaturas
4. Referências
5. Visão geral do restante do documento

## 2. Descrição Geral

1. Perspectiva do produto
2. Funções do produto
3. Características dos usuários
4. Restrições gerais
5. Suposições e dependências

## 3. Requisitos específicos (requisitos funcionais e não-funcionais)

## 4. Índice

## 5. Apêndices.



# EXERCÍCIO – ENADE (I)

Atualmente um número crescente de empresas investe em Tecnologia de Informação (TI), contudo, investir em TI somente por modismo, sem se preocupar com os procedimentos adequados para que isso ocorra, gera um risco elevado. A TI exige qualidade em seu desenvolvimento, principalmente na criação dos sistemas de informações gerenciais, onde a Engenharia de Software deve ser aplicada. O ambiente deve estar preparado para receber esta tecnologia e saber fazer bom uso dela. Em algumas ocasiões, é bastante complexo estabelecer uma forma para mensurar e quantificar a capacidade que a TI possui para criar valores, reduzir custos e avaliar se realmente a empresa está se tornando melhor ou mais competitiva com sua aplicação.

PRADO, A. et al. Tecnologia da Informação Aplicada de Forma Estratégica nos Processos Organizacionais, por Meio de Sistemas de Informações Gerenciais. Janus Revista de Pesquisa Científica - UNIFATEA, Lorena, v. 4, n. 5, p 47-64, jan. – jun. 2007 (adaptado).



# EXERCÍCIO – ENADE

Considerando as informações do texto, avalie as afirmações a seguir, acerca das atividades fundamentais da Engenharia de Software.

- I. A Especificação do Software é a etapa em que clientes e engenheiros projetam e programam o software.
- II. A Validação de Software é a etapa para assegurar se o programa atende as demandas do cliente.
- III. A Evolução do Software é a etapa de modificação para definir a mudança de requisitos, tanto do cliente quanto do mercado.

É correto o que se afirma em:

- a) I, apenas.
- b) III, apenas.
- c) I e II, apenas.
- d) II e III, apenas.
- e) I, II e III



DV0

RESPOSTA: D

DENILCE VELOSO; 2024-03-07T17:56:56.168

# EXERCÍCIO – ENADE (II)

A etapa de definição de requisitos é voltada para estabelecer quais as funções são requeridas pelo sistema e as restrições sobre a operação e o desenvolvimento do software. Os requisitos de software podem ser classificados como ~~requisitos funcionais e não funcionais. SOMMERVILLE, I. Engenharia de Software, 10. ed. São Paulo: Pearson Education, 2019 (adaptado).~~

Considerando as informações do texto, assinale a alternativa em que o item é um requisito funcional.

- A) O software deve ser operacionalizado no sistema Linux.
- B) O tempo de desenvolvimento não deve ultrapassar seis meses.
- C) O software deve emitir relatórios de compras a cada quinze dias.
- D) O tempo de resposta do sistema não deve ultrapassar 30 segundos.
- E) A base de dados deve ser protegida para acesso apenas de usuários autorizados.



DV0

Respostas: C

DENILCE VELOSO; 2024-03-07T18:08:35.214

# ATIVIDADE 4

---



## Exercício Aula 4

**Subir/Upar o arquivo na sua área do GitHub na pasta Atividade4:  
SEUUSUARIO/ES2N/Atividade4**

# PRÓXIMO PASSO PARA O PROJETO (I)



**ATENÇÃO:** O projeto deve ser acompanhado via **GitHub**, incluindo as atividades para fazer, em andamento e entregue.

Itens que serão considerados na avaliação do projeto

- Entrega dos documentos/artefatos **dentro do prazo**
- Participação nas reuniões de **orientação durante a aula**
- Gerenciamento do Projeto no **GitHub**
- **Apresentação final** do projeto (final do semestre)

Lembrando que todo os documentos/artefatos produzidos serão utilizados nas disciplinas ES3 e LES, e no trabalho de graduação final.

Média = (Avaliação Teórica \* 20% + **Projeto \* 60%** + Média Atividades extras \* 20%)



# PRÓXIMO PASSO PARA O PROJETO (II)

---



HOJE

- Discussão sobre a 3ª versão da Proposta de Projeto

# PRÓXIMO PASSO PARA O PROJETO (III)



## NOVAS ENTREGAS

---

1. Nova versão **Proposta ES2N-Proposta v4.0**
2. Fazer 1ª versão Lista de Requisitos Funcionais do Sistema **ES2N-Requisitos Funcionais v1.0**
3. Primeira versão das questões para a pesquisa (**ES2N-AnexoPesquisa v1.0**), dependendo do caso são 2 questionários. Mais ou menos 12 perguntas de alternativas quando for para os possíveis usuários. Deixar uma em aberto para sugestões/críticas.

# PRÓXIMOS PASSOS PARA O PROJETO INTEGRADOR – SEMANA QUE VEM



## Sobre a pesquisa (1):

---

- Faça as perguntas o mais simples possível. Evite perguntas longas ou complexas;
- Use uma linguagem clara e concisa;
- Use opções de resposta limitadas. Isso tornará mais fácil para os respondentes responderem às perguntas.
- Ofereça opções de resposta que sejam relevantes para o público-alvo da pesquisa;

# PRÓXIMOS PASSOS PARA O PROJETO INTEGRADOR – SEMANA QUE VEM



---

## Sobre a pesquisa (2):

Aqui estão alguns exemplos de perguntas que podem ser consideradas cansativas para os usuários:

- Perguntas abertas que exigem respostas longas;
- Perguntas que exigem que os usuários pesquisem informações;
- Perguntas que são repetitivas ou irrelevantes;
- Perguntas que são muito pessoais ou intrusivas.

# REFERÊNCIAS

---

- MACHADO, Felipe Nery Rodrigues. **Análise e gestão de requisitos de software**: onde nascem os sistemas. 1ª. ed. São Paulo: Érica, 2011.
- PRESSMAN, Roger S. **Engenharia de software : uma abordagem profissional**. 7. ed. Porto Alegre: McGraw-Hill, 2011.
- SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo: Pearson, 2019.
- \_\_\_\_\_. Materiais Fornecidos pelo autor. Disponível em:  
<<https://www.slideshare.net/software-engineering-book/ch4-req-eng>> Acesso em:  
01 de mar. de 2021.

