



# ENGENHARIA DE REQUISITOS

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Prof. Renato Matroniani



EDUCAÇÃO  
METODISTA

# Referências

- SOMMERVILLE, Ian. Engenharia de Software. 10ª ed. São Paulo: Pearson Prentice Hall, 2011. (Biblioteca Virtual) <https://plataforma.bvirtual.com.br/Leitor/Publicacao/168127/pdf/0>
- PRESSMAN, R. S. Engenharia de Software. Rio de Janeiro: McGraw Hill, 2002, 704p.
- SBROCCO, J. H.; MACEDO, P. C., Metodologias Ágeis: Engenharia de Software sob Medida., São Paulo: Érica, 2012, 256p.
- CHIKOFFSKY, Elliot. Computer-Aided Software Engineering (CASE). COMPUTER IEEE Computer Society, 1993.

# Requisitos Funcionais e Não Funcionais

- forma de classificação dos requisitos de software
- funcionais: declarações de serviços que o sistema deve fornecer, como deve reagir a determinadas entradas e como o sistema deve se comportar em determinadas situações.
- (em algumas situações, devem explicitar o que o sistema não deve fazer).



# Requisitos Funcionais e Não Funcionais

- não funcionais: são basicamente as **restrições** aos serviços ou funções oferecidos pelo sistema.
- restrições de timing;
- restrições do processo de desenvolvimento;
- restrições de normas.

# Requisitos Funcionais e Não Funcionais

- Um requisito de usuário relacionado com proteção, por exemplo, limitação de acesso, é um requisito funcional ou não funcional? Por que?
- Os requisitos não são independentes, e podem gerar ou restringir outros requisitos!

# Requisitos Funcionais e Não Funcionais

limitação de acesso  
a usuários  
autorizados



necessidade de  
incluir recursos de  
usuários no sistema



# Requisitos Funcionais

- descrevem o que o sistema deve fazer
- dependem do tipo de software a ser desenvolvido
- dependem de quem são os possíveis usuários
- dependem da abordagem geral da organização ao escrever os requisitos.

# Requisitos Funcionais

- podem ser descritos de forma abstrata, para serem entendidos pelos usuários do sistema.
- podem ser gerais, quando descreve de forma abrangente o que o sistema deve fazer.
- podem ser mais específicos e detalhados, quando descrevem as funções do sistema e as formas de trabalho em uma organização.



# Requisitos Funcionais

- voltando ao exemplo da clínica:

## Definição de requisitos de usuário

1. O MHC-PMS deve gerar relatórios gerenciais mensais que mostrem o custo dos medicamentos prescritos por cada clínica durante aquele mês.

## Especificação de requisitos de sistema

- 1.1 No último dia útil de cada mês deve ser gerado um resumo dos medicamentos prescritos, seus custos e as prescrições de cada clínica.
- 1.2 Após 17:30h do último dia útil do mês, o sistema deve gerar automaticamente o relatório para impressão.
- 1.3 Um relatório será criado para cada clínica, listando os nomes dos medicamentos, o número total de prescrições, o número de doses prescritas e o custo total dos medicamentos prescritos.
- 1.4 Se os medicamentos estão disponíveis em diferentes unidades de dosagem (por exemplo, 10 mg, 20 mg), devem ser criados relatórios separados para cada unidade.
- 1.5 O acesso aos relatórios de custos deve ser restrito a usuários autorizados por uma lista de controle de gerenciamento de acesso.

Texto baseado em SUMMERVILLE (2011)



EDUCAÇÃO  
METODISTA

# Requisitos Funcionais

- exemplos de requisitos funcionais para esse caso:
- um usuário deve ser capaz de pesquisar as listas de agendamentos para todas as clínicas;
- o sistema deve gerar a cada dia, para cada clínica, a lista dos pacientes para as consultas daquele dia.
- cada membro da equipe, usuário do sistema, deve ser identificado apenas por seu número de 8 dígitos.

# Requisitos Funcionais

- requisitos funcionais dos usuários definem os recursos específicos a serem fornecidos pelo sistema.
- geralmente são retirados do documento de requisitos de usuário
- mostram que os requisitos funcionais podem ser escritos em diferentes níveis de detalhamento.



# Requisitos Funcionais

- imprecisão na especificação de requisitos causam muitos problemas da engenharia de software.
- requisitos “ambíguos” também causam problemas e interpretações errôneas por parte dos desenvolvedores.
- se isso ocorre, é necessário muitas vezes voltar à fase de elicitação, estabelecer novos requisitos e/ou fazer alterações no sistema → atrasos e custos

# Requisitos Funcionais

- Exemplo das clínicas:
- Um requisito hipotético diz que um usuário deve ser capaz de buscar as listas de agendamento para todas as clínicas.
- O problema desse requisito é que os pacientes podem se confundir.
- a consulta ser em uma clínica, mas se deslocarem até outra.
- O sistema vai registrar como consulta realizada, mesmo em outra clínica.

# Requisitos Funcionais

- O membro da equipe médica que especifica este requisito espera que “pesquisar” significa que dado o nome\_do\_paciente, o sistema vai procurá-lo em todos os \_agendamentos\_ em todas as \_clinicas\_.
- Porém, isso não ficou explícito no requisito!
- Consequências: desenvolvedores podem interpretar o requisito de maneira diferente ->>

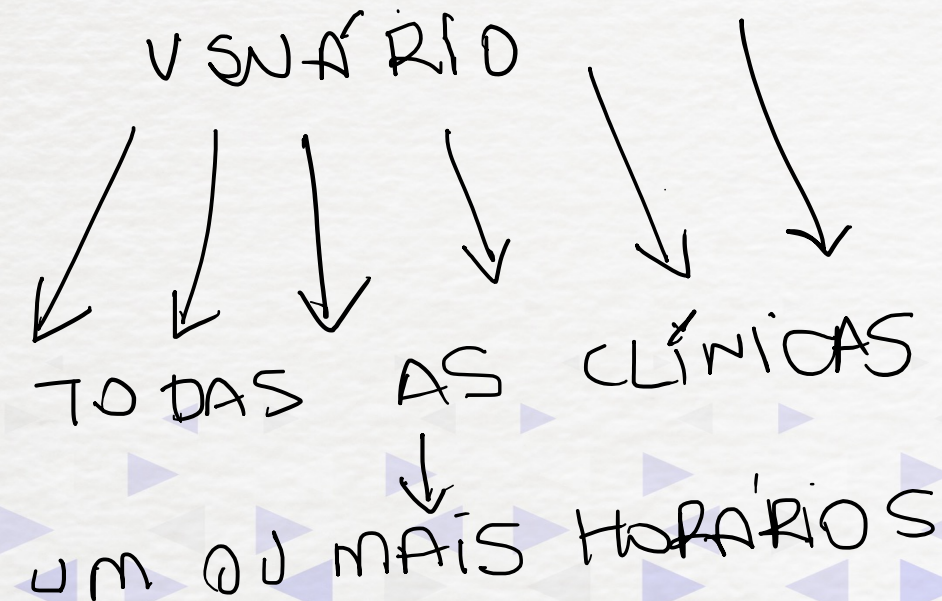


# Requisitos Funcionais

- Eles podem implementar uma pesquisa onde o usuário tenha que escolher uma clínica, e, em seguida, realizar a pesquisa.
- Isso envolverá mais entradas do usuário.
- Mais tempo gasto.

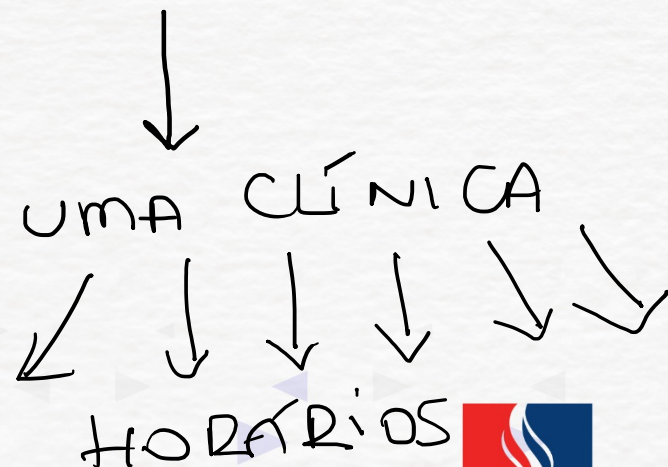
# Requisitos Funcionais

PARA A EMPRESA:



PARA O DEV:

USUÁRIO



Texto baseado em SOMMERVILLE (2011)

# Requisitos Funcionais

- Como então estruturar da melhor forma este caso?





# Requisitos Funcionais

- especificação dos requisitos deve ser completa e consistente.
- completude: todos os serviços solicitados pelo usuário devem ser definidos.
- consistência: requisitos não devem ter definições contraditórias.
- impossível alcançar 100%: erros, omissões, necessidades dos stakeholders.

*Texto baseado em SOMMERVILLE (2011)*

# Requisitos Funcionais

- o problema pode aparecer após o sistema ter sido entregue ao cliente.
- e agora?
- como evitar?

# Requisitos NÃO Funcionais

- são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema aos seus usuários.
- podem estar relacionados às propriedades emergentes do sistema: confiabilidade, tempo de resposta, ocupação de área.



# Requisitos NÃO Funcionais

- também podem **estabelecer restrições** sobre a implementação do sistema, dos dispositivos de E/S, ou as representações da GUI.
- desempenho, proteção e disponibilidade especificam e restringem as características do sistema como um todo.
- são mais críticos que requisitos funcionais individuais.

# Requisitos NÃO Funcionais

- e quando os usuários encontram maneiras de contornar uma função do sistema que não atenda às suas necessidades?

# Requisitos NÃO Funcionais

- deixar de atender um requisito não funcional pode significar a inutilização de todo o sistema.
- exemplo: confiabilidade de sistemas de aeronaves



# Requisitos NÃO Funcionais

- requisitos não funcionais são mais difíceis de identificar.
- requisitos não funcionais podem afetar a arquitetura geral do sistema em vez de apenas componentes individuais.
- um requisito não funcional pode gerar uma série de requisitos funcionais relacionados. Exemplo: requisitos de proteção.

# Requisitos NÃO Funcionais

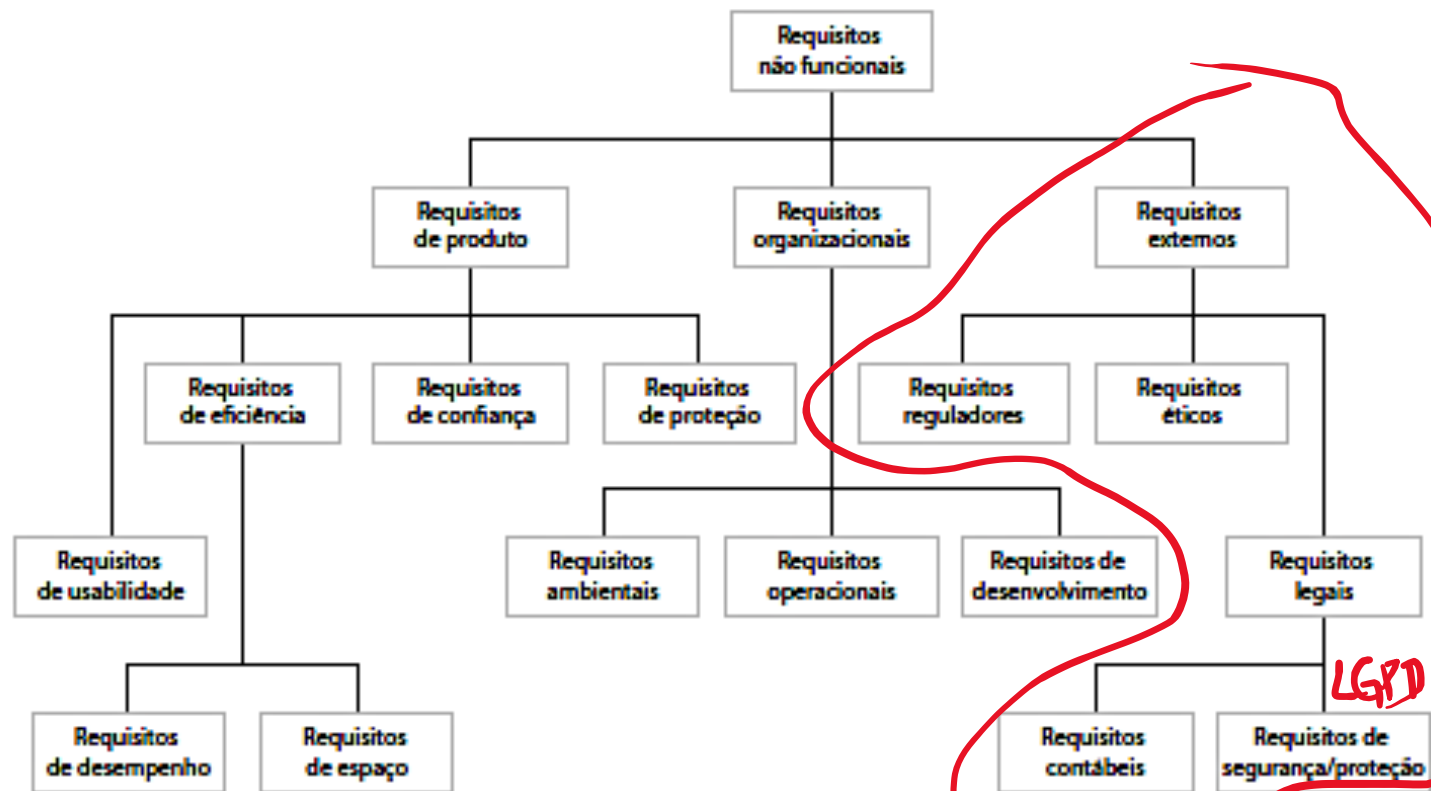
- Quais são as necessidades dos usuários que geram os requisitos não funcionais?

# Requisitos NÃO Funcionais

- Quais são as necessidades dos usuários que geram os requisitos não funcionais?
  - restrições de orçamento
  - políticas organizacionais
  - necessidade de operação com outros sw ou hw
  - fatores externos como regulamentos de segurança e privacidade



# Requisitos NÃO Funcionais



# Requisitos NÃO Funcionais

- Requisitos de produto: especificam e/ou restringem o comportamento do software.
- Exemplos:
  - rapidez necessária para execução de tarefas
  - quanto de memória é necessário
  - confiabilidade – taxa aceitável de falhas [ex. caso Itaú]
  - proteção
  - usabilidade

# Requisitos NÃO Funcionais

- Requisitos organizacionais: derivados de políticas e procedimentos da organização do cliente e do desenvolvedor.
- exemplos:
  - processo operacional
  - processo de desenvolvimento (inclui linguagem de programação)
  - ambiente de desenvolvimento
  - ambiente operacional do sistema

*Texto baseado em SOMMERVILLE (2011)*



**EDUCAÇÃO  
METODISTA**



# Requisitos NÃO Funcionais

- Requisitos externos: derivam de fontes e fatores externos ao sistema e ao desenvolvimento.
- podem incluir requisitos reguladores (definições por normas para serem aprovados para o uso).
- requisitos legais – se o sistema opera dentro da lei.
- requisitos éticos – assegurar que o sistema será aceitável para os usuários e público em geral.

# Requisitos NÃO Funcionais

## Quadro 4.1 Exemplos de requisitos não funcionais no MHC-PMS.

### Requisito de produto

O MHC-PMS deve estar disponível para todas as clínicas durante as horas normais de trabalho (segunda a sexta-feira, 8h30 às 17h30). Períodos de não operação dentro do horário normal de trabalho não podem exceder cinco segundos em um dia.

### Requisito organizacional

Usuários do sistema MHC-PMS devem se autenticar com seus cartões de identificação da autoridade da saúde.

### Requisito externo

O sistema deve implementar as disposições de privacidade dos pacientes, tal como estabelecido no HStan-03-2006-priv.