

# **Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - IFSP Câmpus Jacareí**

**Tecnologia em Análise e Desenvolvimento de Sistemas -  
ADS**

2º Semestre de 2023

**Engenharia de Software 1 – JCRESW1**

Prof. Lineu Mialaret

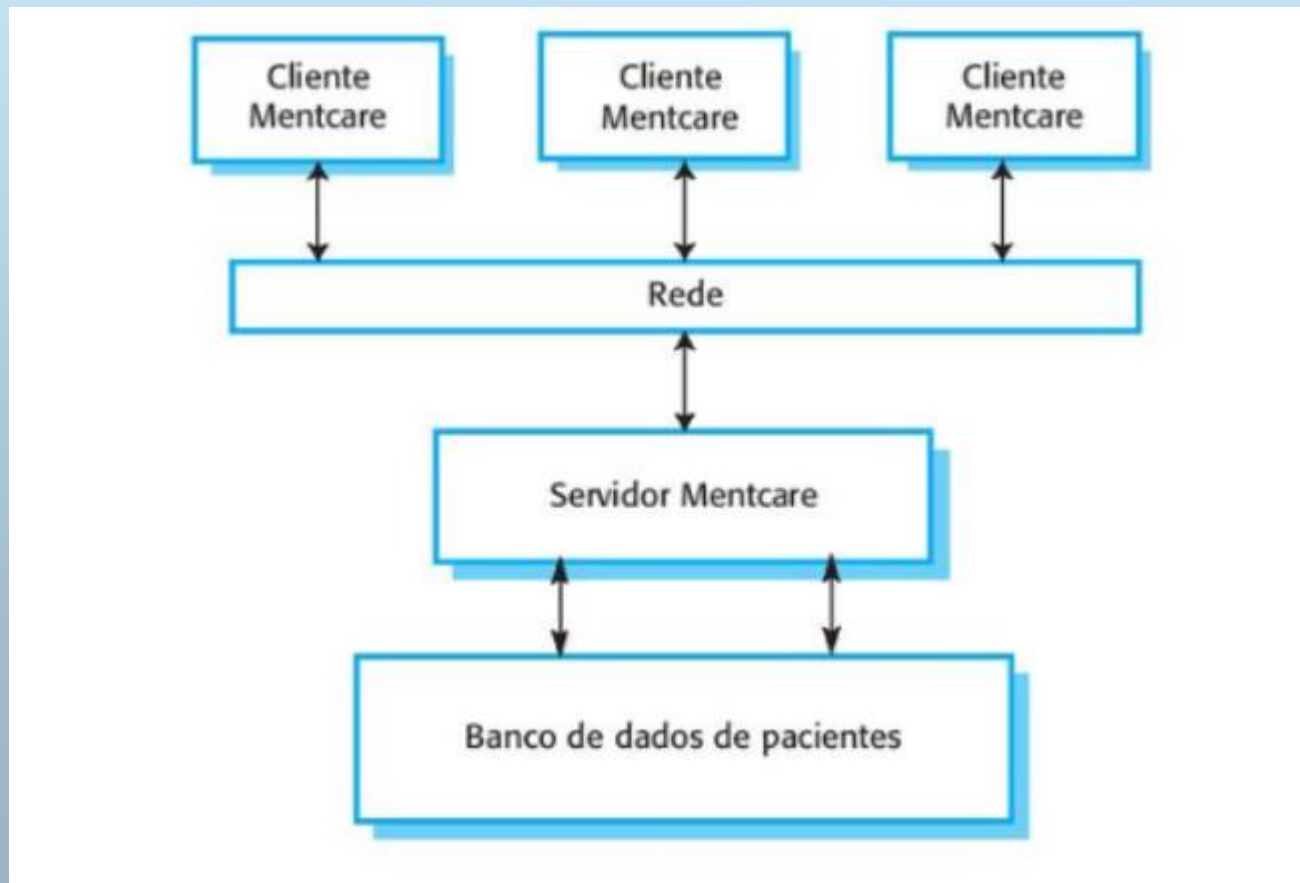
**Aula 8: Engenharia de Requisitos (1)**

# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)

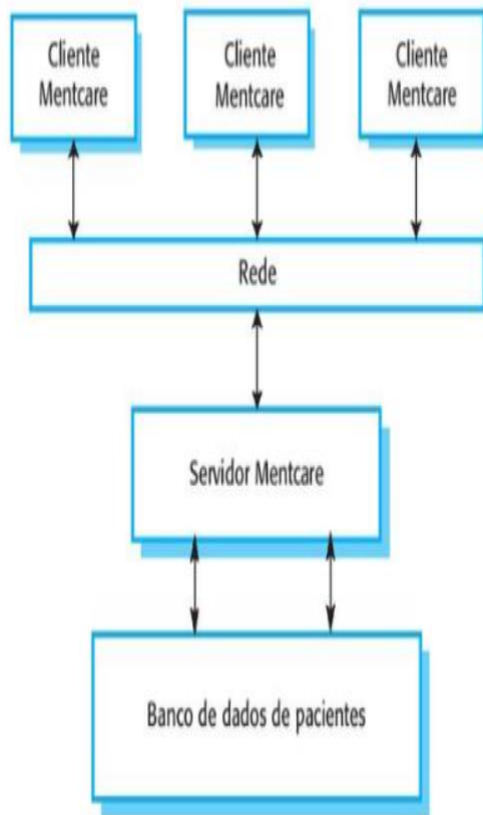
- O Sistema Mentcare é um Sistema de Informação de pacientes que fornece suporte ao cuidado com a saúde mental e mantém dados sobre os pacientes que sofrem de problemas de saúde mental e os tratamentos aos quais foram submetidos.
- A maioria dos pacientes não precisa de tratamento hospitalar dedicado, mas apenas frequentar regularmente uma clínica especializada, onde podem encontrar um médico que tenha conhecimento detalhado de seus problemas.
- Para facilitar a frequência dos pacientes, essas clínicas não existem apenas dentro dos hospitais, mas em postos médicos locais ou centros comunitários.

# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)

- O Sistema Mentcare, com uma possível arquitetura preliminar, é um sistema de informação de pacientes para ser utilizado nessas clínicas.

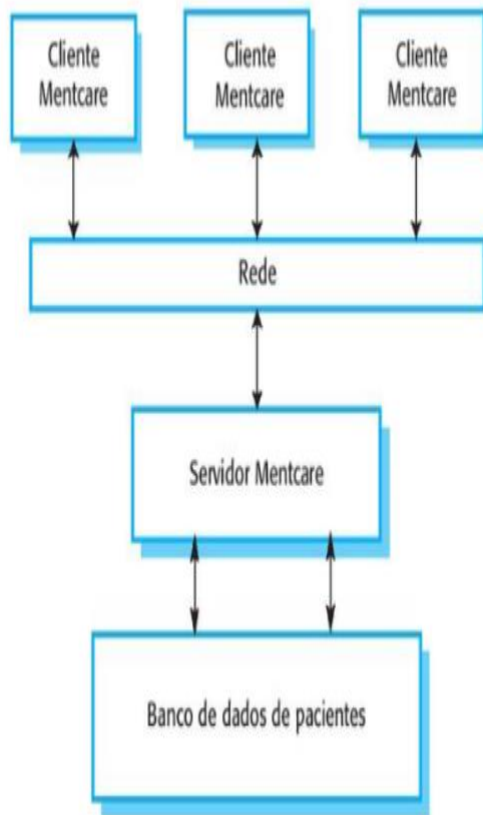


# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)



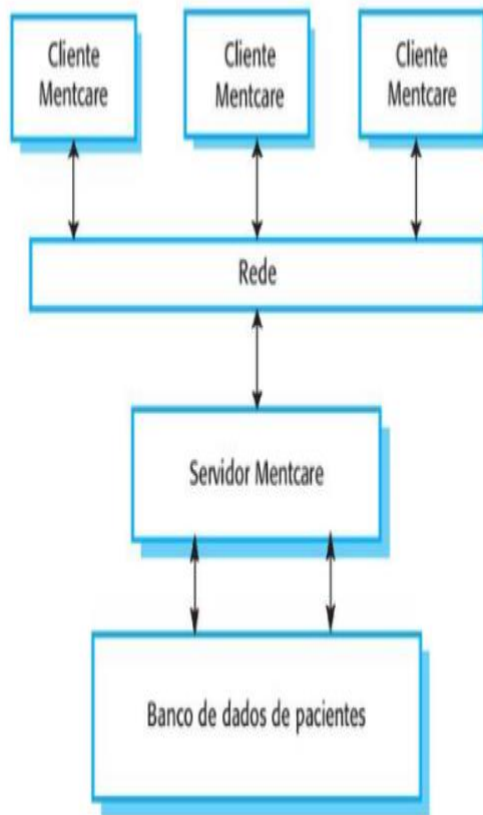
- Ele utiliza um banco de dados centralizado, que contém informações sobre os pacientes, mas também foi concebido para ser executado em um *notebook*, de modo que possa ser acessado e utilizado em locais sem acesso a uma conexão de rede segura.
- Quando os sistemas locais contam com acesso seguro à rede, eles usam as informações de pacientes que constam do banco de dados, mas podem baixar e usar cópias locais dos registros dos pacientes quando estiverem desconectados.

# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)



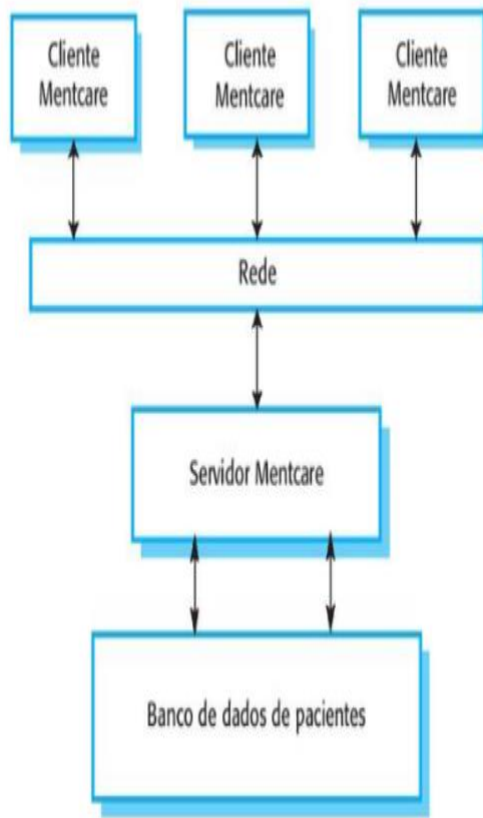
- O sistema não mantém um registro de informações médicas completo e, portanto, não mantém informações sobre outras condições clínicas.
- No entanto, ele pode interagir e trocar dados com outros sistemas de informações médicas.
- Este sistema tem duas finalidades:
  - Gerar informações de gestão que permitam que gestores de serviços de saúde avaliem o desempenho em relação às metas locais e governamentais.
  - Fornecer informações atualizadas à equipe médica para tratamento dos pacientes.

# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)



- Os usuários do sistema incluem pessoal das clínicas, como médicos, profissionais de enfermagem e assistentes de saúde (profissionais de enfermagem que visitam as pessoas em casa para acompanhar o seu tratamento).
- Os usuários que não são da área médica incluem recepcionistas que marcam consultas, profissionais responsáveis pela manutenção do sistema de cadastros e pessoal administrativo que gera os relatórios.

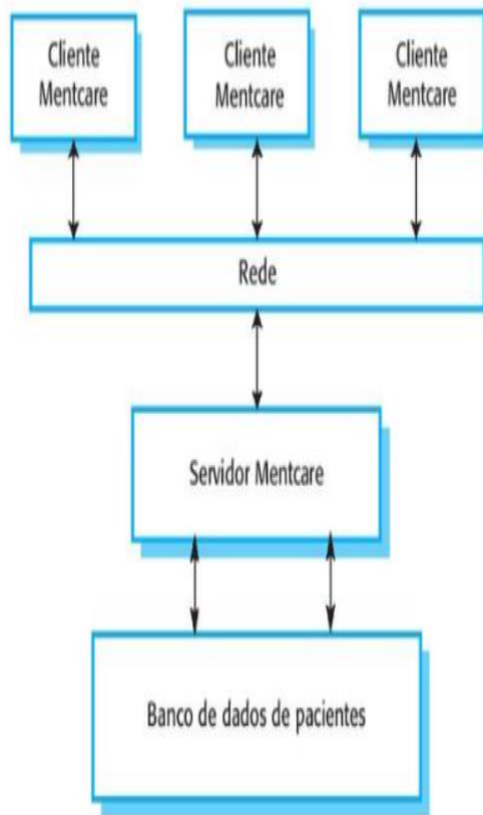
# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)



- O sistema é utilizado para registrar informações sobre os pacientes (nome, endereço, idade, familiares, etc.), consultas (data, médico consultado, impressões subjetivas do paciente, etc.), condições e tratamentos.
- Os relatórios são gerados em intervalos regulares para a equipe médica e os gestores de saúde autorizados.
- Normalmente, os relatórios para a equipe médica se concentram nas informações de cada paciente, enquanto que os relatórios de gestão são anônimos e se preocupam com condições, custos de tratamento, etc.

# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)

- Quanto à gestão do cuidado individual:
  - Os médicos podem criar registros para os pacientes, editar as informações no sistema, visualizar o histórico do paciente, etc.
  - O sistema dá suporte ao resumo de dados, de modo que os médicos que ainda não conheçam o paciente possam aprender rapidamente sobre seus principais problemas e os tratamentos prescritos.

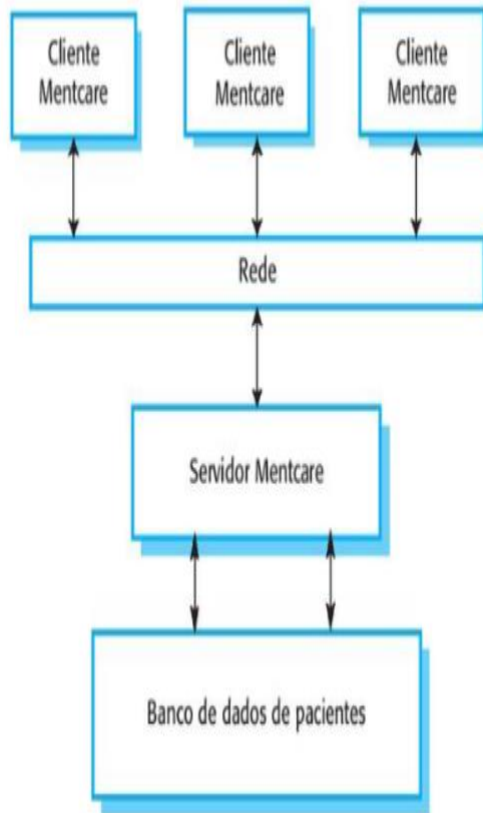




# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)

- Quanto ao monitoramento do paciente:

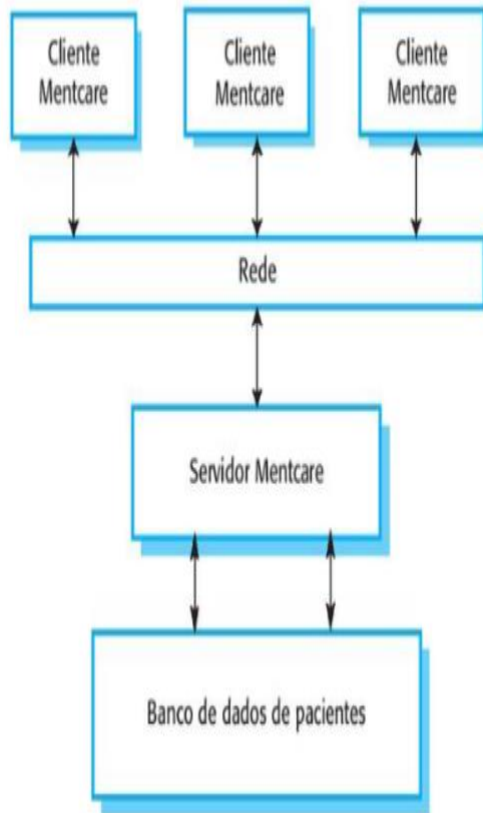
- O sistema monitora regularmente os registros dos pacientes que estão sendo tratados e emite alertas se forem detectados possíveis problemas.
- Portanto, se um paciente não foi atendido por um médico durante certo intervalo de tempo, pode ser emitido um alerta.
- Um dos elementos mais importantes do monitoramento do paciente é acompanhar os pacientes que foram internados e assegurar que sejam realizadas as averiguações exigidas legalmente, no momento certo.



# Sistema de Informação de Pacientes de Saúde Mental (Mentcare)

- Quanto aos relatórios administrativos:

- O sistema gera relatórios mensais de gestão, mostrando a quantidade de pacientes tratados em cada clínica, a quantidade de pacientes que entrou e saiu do sistema de atendimento, a quantidade de pacientes internados, os medicamentos prescritos e seus custos, dentre outras informações disponíveis.



# Engenharia de Requisitos

## ■ Requisito:

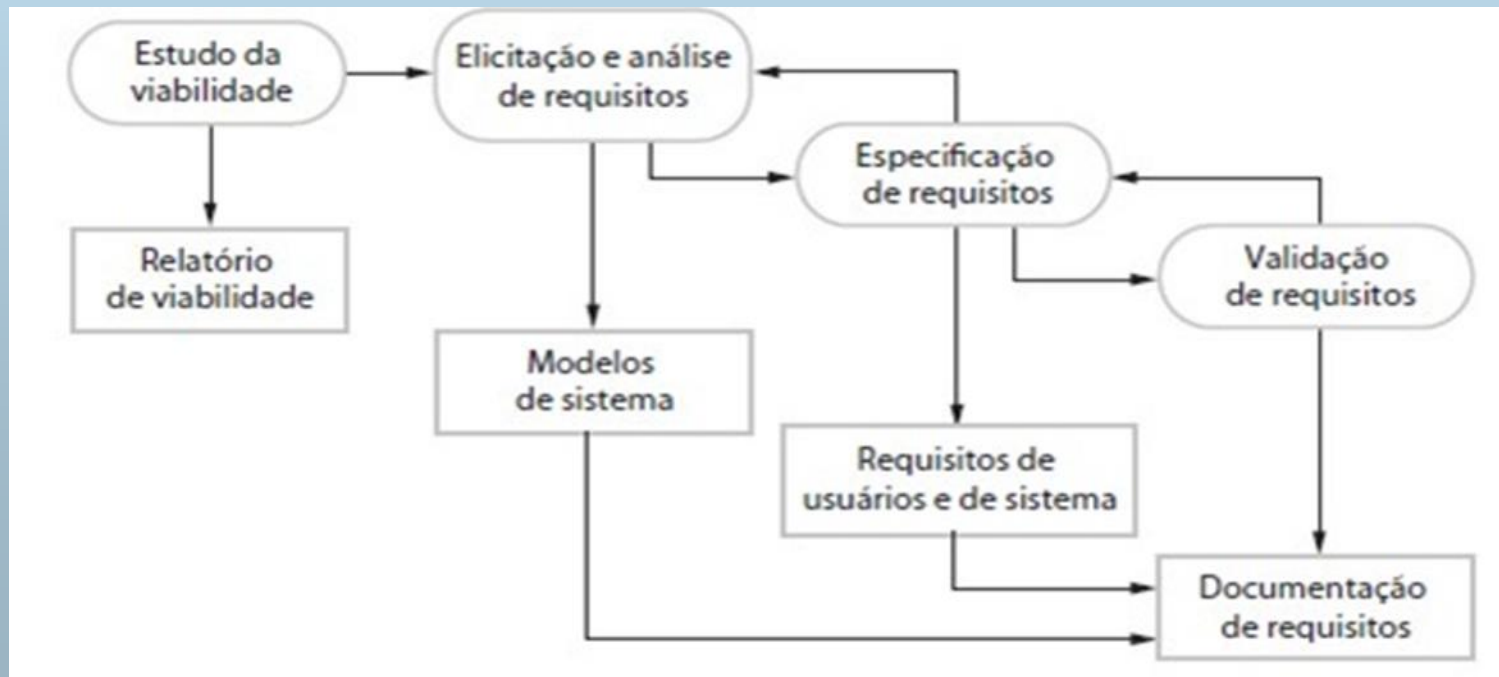
- Alguma coisa requerida, desejada ou necessária. [Webster's Ninth New Collegiate Dictionary]
- Condição ou capacidade necessária a um usuário para resolver um problema ou alcançar um objetivo.
- Condição ou capacidade que deve ser alcançada ou possuída por um sistema ... para satisfazer um contrato, padrão, especificação ou qualquer outro documento formal. [IEEE Standard 729]

## ■ Especificação:

- Descrição minuciosa das características que um material, obra, ou serviço deverão apresentar. [AURÉLIO]

# Engenharia de Requisitos (cont.)

- A Engenharia de Requisitos (ER):
  - Trata do processo de definir, especificar, validar, documentar, conferir e manter requisitos de clientes no processo de desenvolvimento de software.
  - É uma atividade comum na Engenharia de Software.



# Requisitos

- Os requisitos de um sistema são as descrições dos serviços (funcionalidades) que o sistema deve prestar e as restrições a sua operação.
  - Esses requisitos refletem as necessidades dos clientes de um sistema que atende a um determinado propósito, como controlar um dispositivo, fazer um pedido ou encontrar informações.
- O termo requisito não é utilizado consistentemente na indústria de software.
  - Em alguns casos, um requisito é simplesmente uma declaração abstrata de alto nível de um serviço que um sistema deve oferecer ou de uma restrição a um sistema.
  - No outro extremo, é uma definição formal detalhada de uma função do sistema.

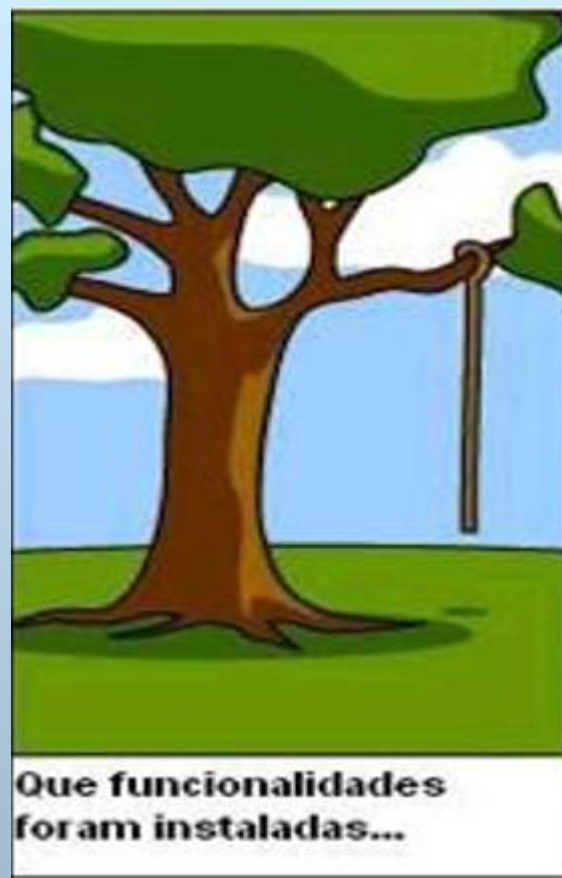
# Requisitos (cont.)

*Se uma empresa deseja assinar um contrato para um grande projeto de desenvolvimento de software, ela deve definir suas necessidades de uma maneira suficientemente abstrata para que não haja uma solução predefinida. Os requisitos devem ser escritos de modo que vários concorrentes possam disputar o contrato, oferecendo, talvez, maneiras diferentes de satisfazer as necessidades da empresa cliente. Depois de assinado o contrato, o contratado deve escrever uma definição mais detalhada do sistema para o cliente, de modo que ele entenda e valide o que o software fará. Esses dois documentos podem ser reunidos em um documento de requisitos do sistema<sup>1</sup>.*

Alguns dos problemas que surgem durante o processo de Engenharia de Requisitos são consequência de não separar claramente os diferentes níveis de descrição.



## Requisitos (cont.)



Alguns dos problemas que surgem durante o processo de Engenharia de Requisitos são consequência de não separar claramente os diferentes níveis de descrição.

# O que é um Requisito?

- It may range from:
  - a high-level abstract statement of a service (e.g. renting service) or,
  - a high-level abstract statement of a system constraint (e.g. authorized people can see patient records) to,
  - a detailed mathematical functional specification (e.g. BMI)
- Requirements are basis of your contract.

$$\text{BMI} = \frac{m}{h^2}$$

BMI = body mass index

$m$  = mass (in kilograms)

$h$  = height (in meters)

The image shows a web form titled "BMI calculator". It has two tabs: "Adult" (selected) and "Child". The form contains several input fields and sections:

- Height:** Two input fields for "Feet" and "Inches", with a link "Switch to cm".
- Weight:** Two input fields for "Stone" and "Pounds", with a link "Switch to kg".
- Age:** A single input field.
- Sex:** Radio buttons for "Male" and "Female", with a link "Why are we asking?".
- Ethnic group (optional):** A dropdown menu currently showing "Not stated", with a link "Why are we asking?".
- Activity level:** Radio buttons for "Inactive" (Less than 30 minutes a week), "Moderately active" (Between 30 and 60 minutes a week), and "Active" (Between 60 and 150 minutes a week), with a link "What counts?".
- Buttons:** A green "Calculate" button and a blue "Reset" link.

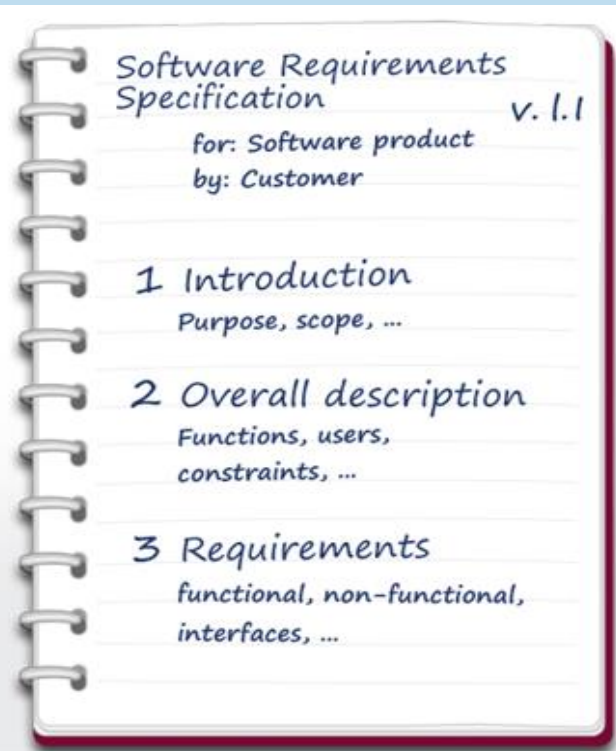


# Requisitos de Usuário

- Requisitos de Usuário são declarações (requisitos abstratos de alto nível), em uma linguagem natural (somada a diagramas e tabelas), dos serviços (funcionalidades) que se espera que o sistema forneça para os usuários e das limitações sob as quais ele deve operar.
  - Esses requisitos podem variar de declarações amplas das características necessárias do sistema até descrições precisas e detalhadas da sua funcionalidade.
  - Esses requisitos são escritos para clientes.

# Requisitos de Sistema

- Os Requisitos de Sistema são descrições mais detalhadas das funções, dos serviços e das restrições operacionais do sistema de software.



- O documento de requisitos de sistema, chamado de Especificação Funcional ou Especificação de Requisitos de Software (*Software Requirements Specification*) deve definir exatamente o que deve ser implementado.
- É um documento estruturado que envolve a descrição detalhada das funcionalidades do sistema, serviços e restrições operacionais.
- Pode fazer parte do contrato entre o adquirente do sistema e os desenvolvedores de software.

# Requisitos de Sistema (cont.)

- São necessários diferentes tipos de requisitos para transmitir as informações a respeito de um sistema para diferentes tipos de leitores.
  - A figura a seguir ilustra a distinção entre requisitos de usuário e de sistema.

## Definição de requisitos de usuário

1. O sistema Mentcare deve gerar relatórios de gestão mensais, mostrando o custo dos medicamentos prescritos por cada clínica naquele mês.

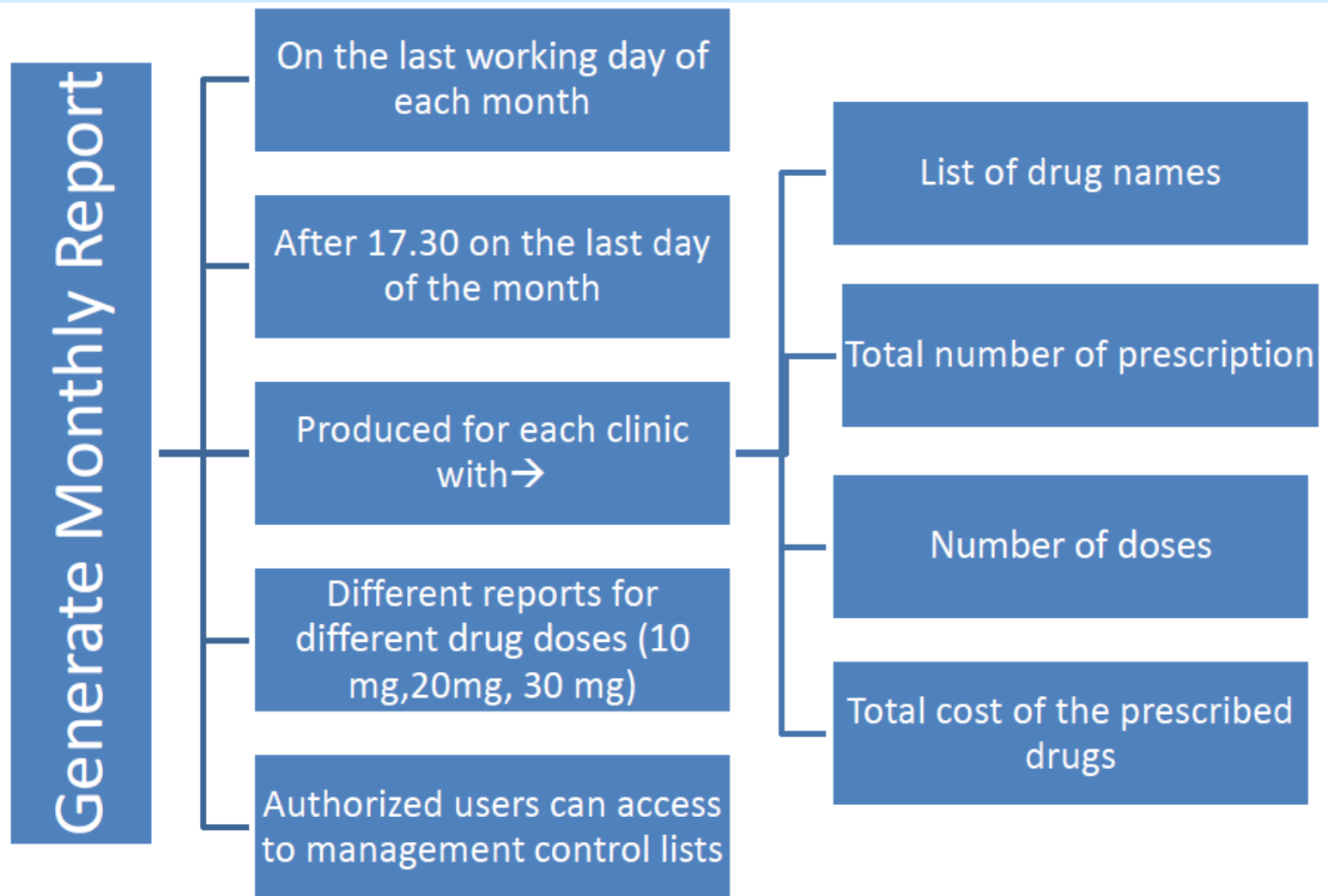
genérico

## Especificação dos requisitos de sistema

- 1.1 No último dia útil de cada mês, deve ser gerado um resumo dos medicamentos prescritos, seu custo e a clínica que os prescreveu.
- 1.2 O sistema deve gerar o relatório para impressão após as 17h30 do último dia útil do mês.
- 1.3 Deve ser criado um relatório para cada clínica, listando o nome de cada medicamento, a quantidade total de prescrições, a quantidade de doses prescritas e o custo total dos medicamentos prescritos.
- 1.4 Se os medicamentos estiverem disponíveis em dosagens diferentes (por exemplo, 10 mg, 20 mg etc.) devem ser criados relatórios diferentes para cada dosagem.
- 1.5 O acesso aos relatórios de medicamentos deve ser restrito aos usuários autorizados, conforme uma lista de controle de acesso produzida pela gestão.

específico

# Requisitos de Sistema (cont.)

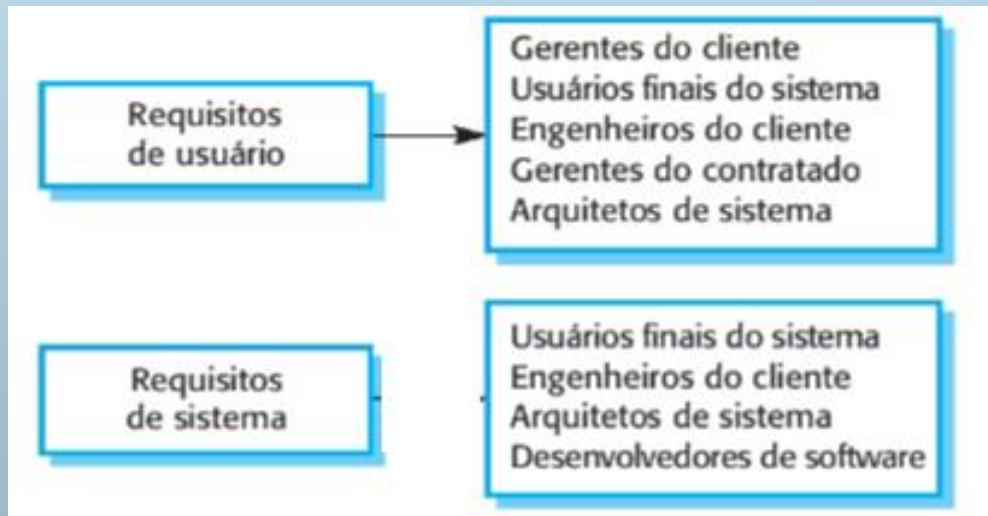


User

System Requirements (SRS)

# Requisitos de Sistema (cont.)

- É necessário escrever os requisitos em diferentes níveis de detalhe, pois diferentes tipos de leitores utilizam esses dados de diferentes maneiras.
  - A figura a seguir mostra os tipos de leitores dos requisitos de usuário e de sistema.

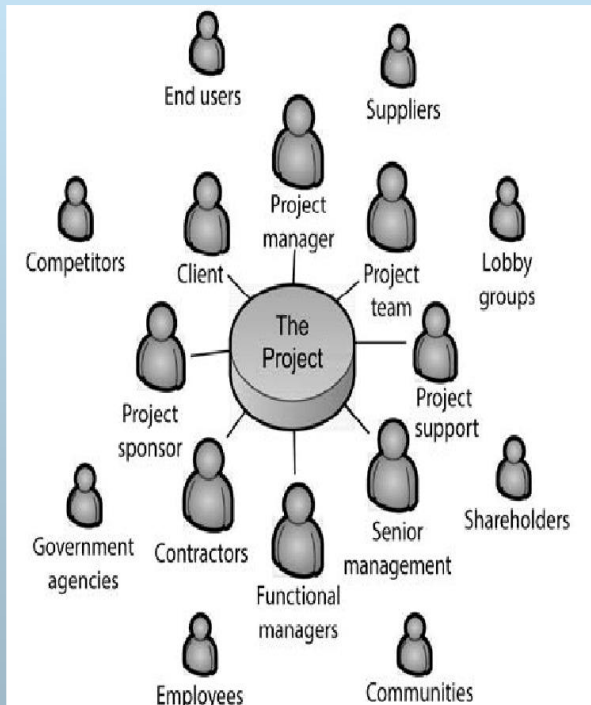


O primeiro grupo geralmente não está preocupado com o modo como o sistema será implementado, e pode ser composto por gerentes que não estejam interessados nos recursos detalhados do sistema.

O segundo grupo precisa saber com maior precisão o que o sistema fará, seja porque estão interessados em saber como ele apoiará os processos da empresa ou porque estão envolvidos na sua implementação.

# Stackholders

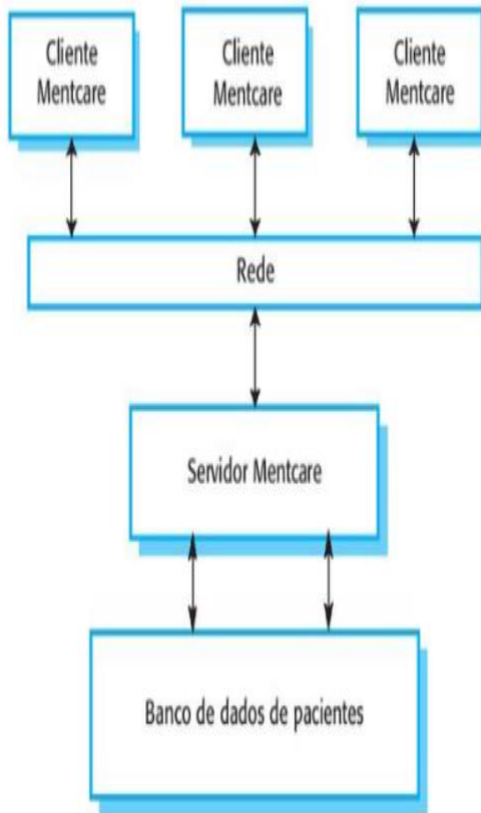
- Assim como os usuários, muitas outras pessoas têm algum tipo de interesse no sistema.
- Os *stakeholders* incluem qualquer um que seja afetado de alguma maneira pelo sistema e, portanto, tenha um interesse legítimo nele.
  - Podem variar de usuários finais de um sistema a gerentes e *stakeholders* externos, como autoridades reguladoras, que certificam a aceitabilidade do sistema.



# Stackholders (cont.)

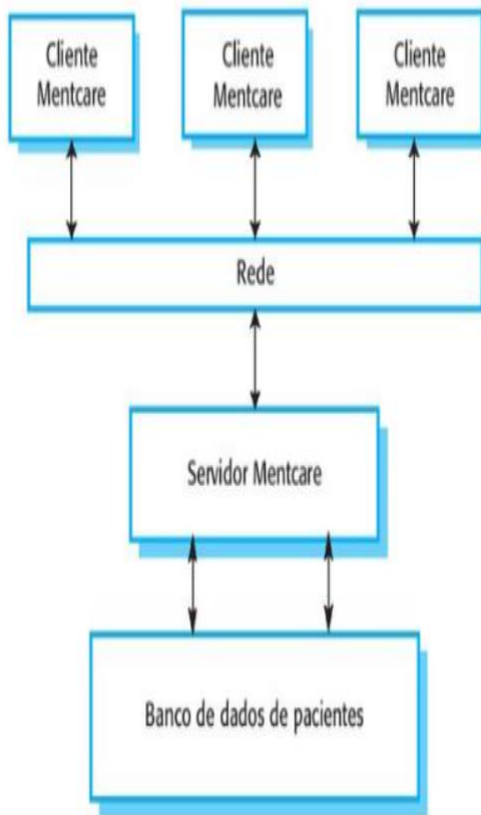
## ■ *Stackholders* do Sistema Mentcare:

- Pacientes cujas informações estão registradas no sistema e familiares desses pacientes.
- Médicos responsáveis por avaliar e tratar os pacientes.
- Profissionais de enfermagem que coordenam as consultas com os médicos e administram alguns tratamentos.
- Recepcionistas que marcam as consultas dos pacientes.
- Equipe de TI responsável pela instalação e manutenção do sistema.
- Gestor de ética médica que deve assegurar que o sistema satisfaz as diretrizes éticas atuais de cuidados com os pacientes.



## Stackholders (cont.)

- Gestores de cuidados com a saúde que obtêm informações gerenciais do sistema.
- Pessoal de controle do prontuário responsável por garantir que as informações do sistema possam ser mantidas e preservadas e que os procedimentos de manutenção de registros tenham sido adequadamente implementados.





# Estudo de Viabilidade

- A Engenharia de Requisitos normalmente é apresentada como o primeiro estágio do processo de desenvolvimento de software.
- Pode ser necessário desenvolver algum nível de compreensão dos requisitos de sistema antes de tomar a decisão de adquirir ou desenvolver um sistema.
  - Essa compreensão inicial dos requisitos estabelece uma visão de alto nível do que o sistema poderia fazer e dos benefícios que poderia proporcionar.
  - Esses aspectos podem ser considerados em um estudo de viabilidade, ferramenta usada para avaliar se o sistema é tecnicamente e financeiramente viável.
  - Os resultados desse estudo ajudam a gestão a decidir se deve ou não seguir adiante com a aquisição ou com o desenvolvimento do sistema.

# Estudo de Viabilidade (cont.)

## Estudos de viabilidade

---

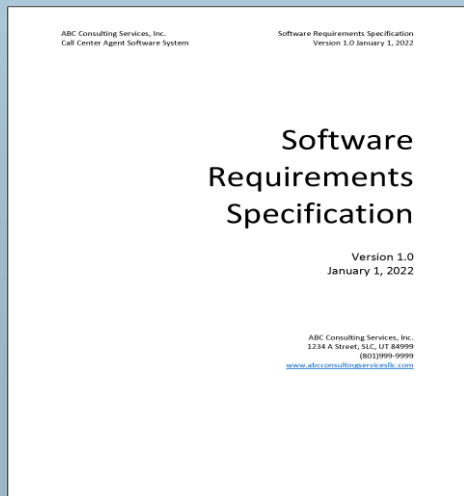
O estudo de viabilidade é um estudo curto e focalizado que deve ser feito no início do processo de ER. Ele deve responder três perguntas fundamentais:

1. O sistema contribui para os objetivos globais da organização?
2. O sistema pode ser implementado dentro do cronograma e orçamento usando a tecnologia atual?
3. O sistema pode ser integrado com outros sistemas utilizados?

Se a resposta a qualquer uma dessas perguntas for não, provavelmente não se deve prosseguir com o projeto.

# Estudo de Viabilidade (cont.)

- Na maioria dos sistemas extensos e complexos há uma fase de engenharia de requisitos claramente identificável antes de começar a implementação do sistema.
  - O resultado é um documento de requisitos, que pode fazer parte do contrato de desenvolvimento do sistema.
  - Naturalmente, são feitas mudanças subsequentes nos requisitos de usuário, que podem ser ampliados para requisitos de sistema mais detalhados.



# Requisitos Funcionais e Não Funcionais



- Os requisitos de sistemas de software são classificados frequentemente como Requisitos Funcionais (RF) ou Requisitos Não Funcionais (RNF).
- Requisitos Funcionais representam as declarações dos serviços (funcionalidades) que o sistema deve fornecer, do modo como o sistema deve reagir a determinadas entradas e de como deve se comportar em determinadas situações.
  - Em alguns casos, os requisitos funcionais também podem declarar explicitamente o que o sistema não deve fazer.

# Requisitos Funcionais e Não Funcionais (cont.)

- Requisitos Não Funcionais representam as restrições sobre os serviços ou funções oferecidas pelo sistema.
  - Eles incluem restrições de tempo, restrições sobre o processo de desenvolvimento e restrições impostas por padrões.
  - Os requisitos não funcionais se aplicam, frequentemente, ao sistema como um todo, em vez de às características individuais ou aos serviços.



# Requisitos Funcionais e Não Funcionais (cont.)

## FUNCTIONAL REQUIREMENTS

WHAT THE SYSTEM SHOULD DO



PRODUCT FEATURES



USER REQUIREMENTS

## NON-FUNCTIONAL REQUIREMENTS

HOW THE SYSTEM SHOULD DO IT



PRODUCT PROPERTIES



USER EXPECTATIONS

# Requisitos Funcionais e Não Funcionais (cont.)

- Requisitos de Domínio representam as restrições sobre sistema relativas ao domínio em que ele opera.
  - São os requisitos do ambiente em que o sistema opera.
- Exemplo:

- 1. Deve haver uma interface-padrão como usuário para todos os banco de dados, que terá como base o padrão Z39.50.
- 2. Em razão das restrições referentes a direitos autorais, alguns documentos devem ser excluídos imediatamente ao serem fornecidos. Dependendo dos requisitos dos usuários, esses documentos serão impressos localmente no servidor do sistema para serem encaminhados manualmente ao usuário ou direcionados para uma impressora de rede.

# Requisitos Funcionais e Não Funcionais (cont.)

- A distinção entre os diferentes tipos de requisitos não é tão clara quanto sugerem essas definições simples.
  - Um requisito de usuário relacionado à segurança da informação (*security*), como uma declaração que limita o acesso aos usuários autorizados, pode parecer um requisito não funcional.
  - Quando desenvolvido em mais detalhes, esse requisito pode gerar outros requisitos claramente funcionais, como a necessidade de incluir no sistema alguns recursos de autenticação do usuário.





# Requisitos Funcionais e Não Funcionais (cont.)


- Isso mostra que os requisitos não são independentes e que, frequentemente, um requisito gera ou limita outros.
  - Portanto, os requisitos de sistema especificam não apenas os serviços ou características, mas também a funcionalidade necessária para garantir que esses serviços/características sejam entregues corretamente.


# Safety and Security


- **Safety** and **Security** are two essential aspects of systems and software.

 **Safety** is the freedom from unacceptable risk or harm. Safety is generally thought of in terms of data integrity. Backups, checksums, etc all ensure that the data is safe from failure.

 Safety deals with preventing accidents, for example using backups against data loss.

 Safety is about the risk of causing damage, to data, to things or to people.

 **Security** may be defined as the prevention of illegal or unwanted interference with the intended operation or inappropriate access of a system.

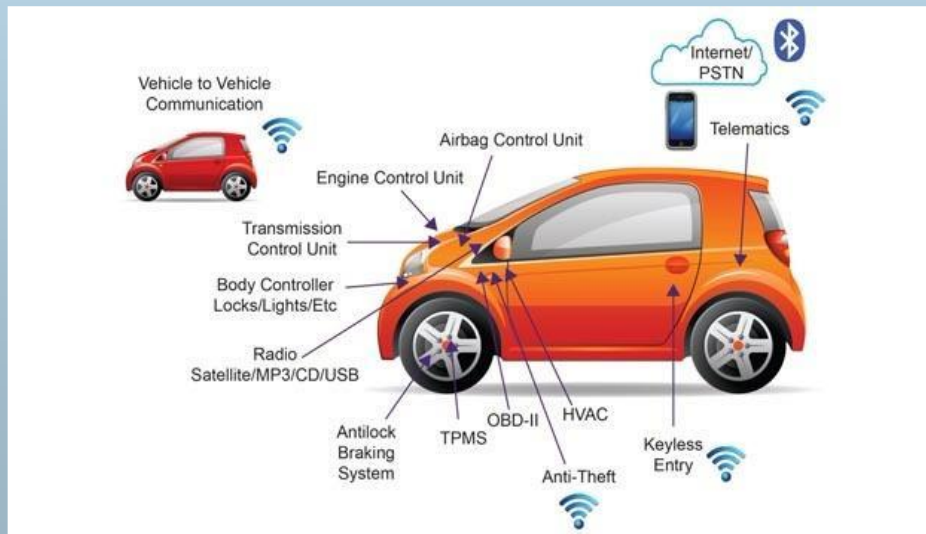
 Security is about ensuring our software functions correctly while under attack and we do this with great measures, by practicing good programming techniques such as validating input from untrusted data.

# Safety

👉 **Safety** is the freedom from unacceptable risk or harm. Safety is generally thought of in terms of data integrity. Backups, checksums, etc all ensure that the data is safe from failure.

📄 Safety deals with preventing accidents, for example using backups against data loss.

📄 Safety is about the risk of causing damage, to data, to things or to people.



# Security

👉 **Security** may be defined as the prevention of illegal or unwanted interference with the intended operation or inappropriate access of a system.

📄 Security is about ensuring our software functions correctly while under attack and we do this with great measures, by practicing good programming techniques such as validating input from untrusted data.



# Security (cont.)

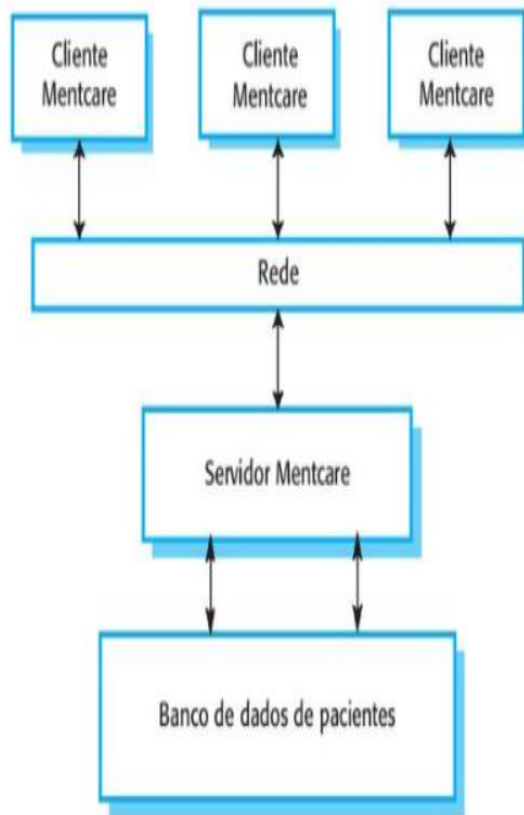
- **Security** refers to activities and measures to ensure the **confidentiality**, **integrity**, and **availability** of an information system and its main asset, data.
- It is important to understand that securing data requires a comprehensive, company-wide approach.
  - 👉 That is, you cannot secure data if you do not secure all the processes and systems around it.
  - 👉 Indeed, securing data entails securing the overall information system architecture, including hardware systems, software applications, the network and its devices, people (internal and external users), procedures, and the data itself.

# Requisitos Funcionais

- Os Requisitos Funcionais de um sistema descrevem o que ele deve fazer e dependem do tipo de software que está sendo desenvolvido, dos usuários esperados para o software e da abordagem geral adotada pela organização ao escrever os requisitos.
  - Quando são apresentados como requisitos de usuário, os requisitos funcionais devem ser escritos de modo compreensível para os usuários e gerentes do sistema.
  - Os requisitos funcionais do sistema expandem os requisitos de usuário e são escritos para os desenvolvedores.
  - Requisitos funcionais devem descrever em detalhes as funções do sistema, suas entradas, saídas e exceções.

# Requisitos Funcionais (cont.)

- Exemplos de requisitos funcionais do Sistema Mentcare, utilizado para manter informações sobre pacientes recebendo tratamento para problemas de saúde mental:



- RF1. Um usuário deve poder fazer uma busca na lista de consultas de todas as clínicas.
- RF2. O sistema deve gerar, a cada dia e para cada clínica, uma lista de pacientes que devam comparecer às consultas naquele dia.
- RF3. Cada membro da equipe que utiliza o sistema deve ser identificado exclusivamente por seu número de funcionário de oito dígitos.

# Requisitos Imprecisos

- A imprecisão na especificação de requisitos pode levar a conflitos entre clientes e desenvolvedores de software.
  - É normal que um desenvolvedor de sistemas interprete um requisito ambíguo de uma forma que simplifique a sua implementação.
    - ❖ Muitas vezes, porém, não é isso o que o cliente quer.
  - Novos requisitos devem ser estabelecidos e mudanças devem ser feitas, o que resulta em atraso na entrega do sistema e aumento dos custos.





# Requisitos Imprecisos (cont.)

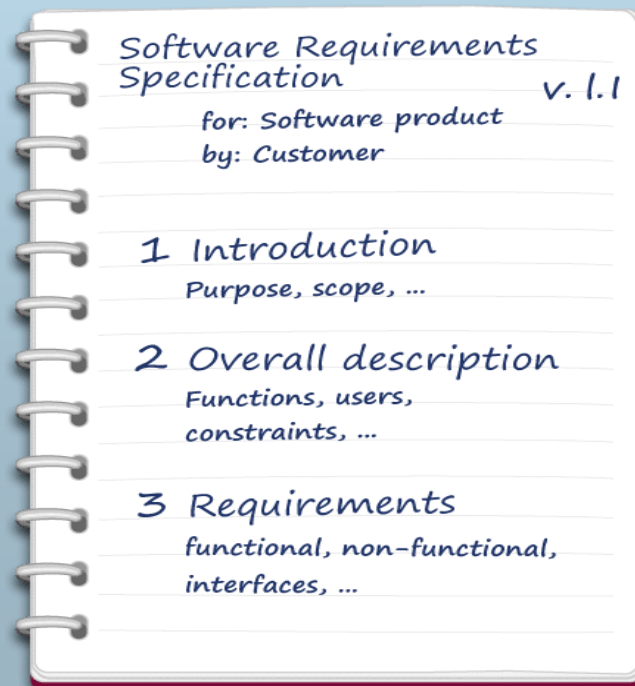
- Exemplo 1:
  - O requisito RF1 do Sistema Mentcare afirma que um usuário deve ser capaz de fazer uma busca nas listas de consultas de todas as clínicas.
    - ❖ O que justifica esse requisito é que os pacientes com transtornos de saúde mental às vezes se confundem.
    - ❖ Eles podem ter uma consulta em uma clínica, mas acabar indo para outra clínica.
    - ❖ Se tiverem uma consulta marcada, serão registrados como atendidos, independentemente da clínica em que estão.

# Requisitos Imprecisos (cont.)

- Exemplo 2:
  - Um membro da equipe médica, ao especificar um requisito de busca, pode esperar que ‘pesquisar’ signifique que, dado o nome de um paciente, o sistema procure por ele em todas as consultas de todas as clínicas.
  - No entanto, isso não está explícito.
    - ❖ Os desenvolvedores de sistemas podem interpretar o requisito do modo mais fácil de implementar.
    - ❖ Sua função de busca pode exigir que o usuário escolha uma clínica e depois faça a pesquisa dos pacientes que compareceram a ela.
    - ❖ Isso envolve mais informações fornecidas pelo usuário e leva mais tempo para completar a busca.

# Requisitos Funcionais (cont.)

- Em condições ideais, a especificação de requisitos funcionais de um sistema deve ser:
  - Completa - todos os serviços e informações requisitados pelo usuário devem estar definidos.
  - Coerente - os requisitos não devem ser contraditórios.



# Requisitos Funcionais (cont.)

- Na prática:
  - Só é possível alcançar a coerência e a completude dos requisitos em sistemas de software pouco extensos (pequenos).
    - ❖ É mais fácil cometer erros e omissões quando se escreve especificações de sistemas extensos e complexos.
  - Sistemas extensos possuem muitos *stakeholders*, com diferentes formações e expectativas, e que tendem a ter necessidades diferentes — e, muitas vezes, inconsistentes.
    - ❖ Essas inconsistências podem não ser óbvias quando os requisitos são especificados em um primeiro momento, e os requisitos inconsistentes podem ser descobertos somente após uma análise mais profunda ou durante o desenvolvimento do sistema.

# Requisitos Não Funcionais

- Os Requisitos Não Funcionais, são aqueles que não possuem relação direta com os serviços específicos fornecidos pelo sistema aos seus usuários.
  - Esses requisitos não funcionais normalmente especificam ou restringem as características do sistema como um todo.
  - Eles podem estar relacionados a propriedades emergentes do sistema, como confiabilidade, tempo de resposta e uso da memória, dentre outras.
  - Podem definir restrições à implementação do sistema, como a capacidade dos dispositivos de E/S ou as representações dos dados utilizados nas interfaces com outros sistemas.

# Requisitos Não Funcionais (cont.)

- Os Requisitos Não Funcionais podem ser mais críticos do que os requisitos funcionais individuais.
  - Os usuários do sistema normalmente encontram maneiras de contornar uma funcionalidade do sistema que não satisfaça suas necessidades.
  - No entanto, descumprir um requisito não funcional pode significar a inutilização total do sistema.
- Exemplos:
  - Se um sistema aviônico não satisfizer seus requisitos de confiabilidade, este não será certificado como seguro para operação.
  - Se um sistema de controle embarcado não cumprir seus requisitos de desempenho, as funções de controle não irão funcionar corretamente.

# Requisitos Não Funcionais (cont.)

- A implementação de requisitos não funcionais pode estar espalhada por todo o sistema devido ao fato desses requisitos não funcionais poderem afetar a arquitetura geral de um sistema em vez de seus componentes individuais.
- Exemplo:
  - Para garantir que sejam cumpridos os requisitos de desempenho em um sistema embarcado, pode ser necessário estruturá-lo em uma dada arquitetura, a fim de minimizar a comunicação entre seus componentes.

# Requisitos Não Funcionais (cont.)

- Exemplo:
  - Um requisito não funcional individual, como um requisito de segurança da informação (*security*), pode gerar vários requisitos funcionais relacionados que definem novos serviços do sistema que se fazem necessários caso o requisito não funcional seja implementado.
  - Também pode-se gerar requisitos que restringem outros requisitos existentes, como por exemplo, poder limitar o acesso à informação no sistema.



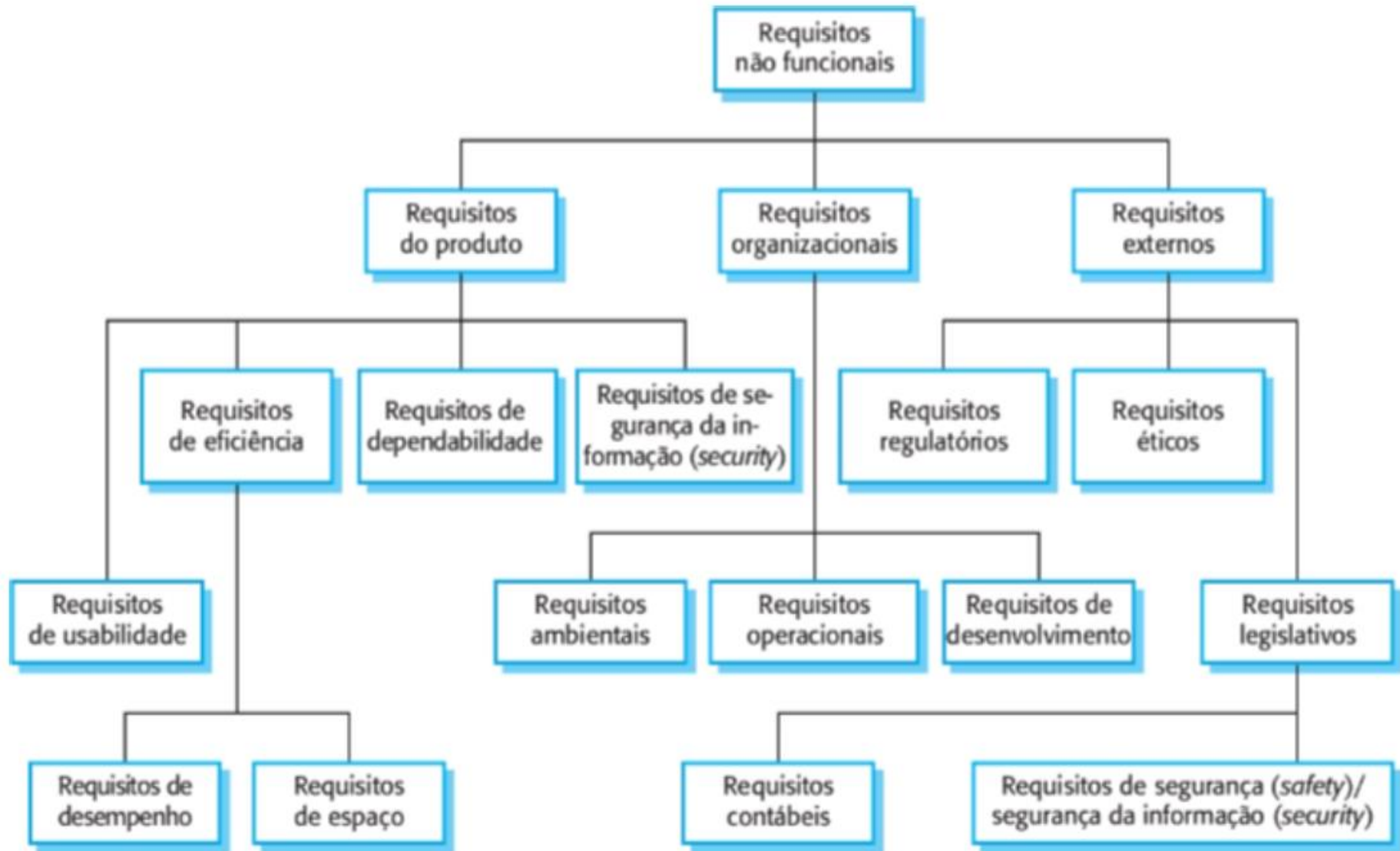


# Requisitos Não Funcionais (cont.)

- Os Requisitos Não Funcionais surgem das necessidades dos usuários, que se devem a:
  - Restrições orçamentárias.
  - Políticas organizacionais.
  - Necessidade de interoperabilidade com outros sistemas de software ou hardware.
  - Fatores externos, como normas de segurança (*safety*) ou legislação relativa à privacidade.

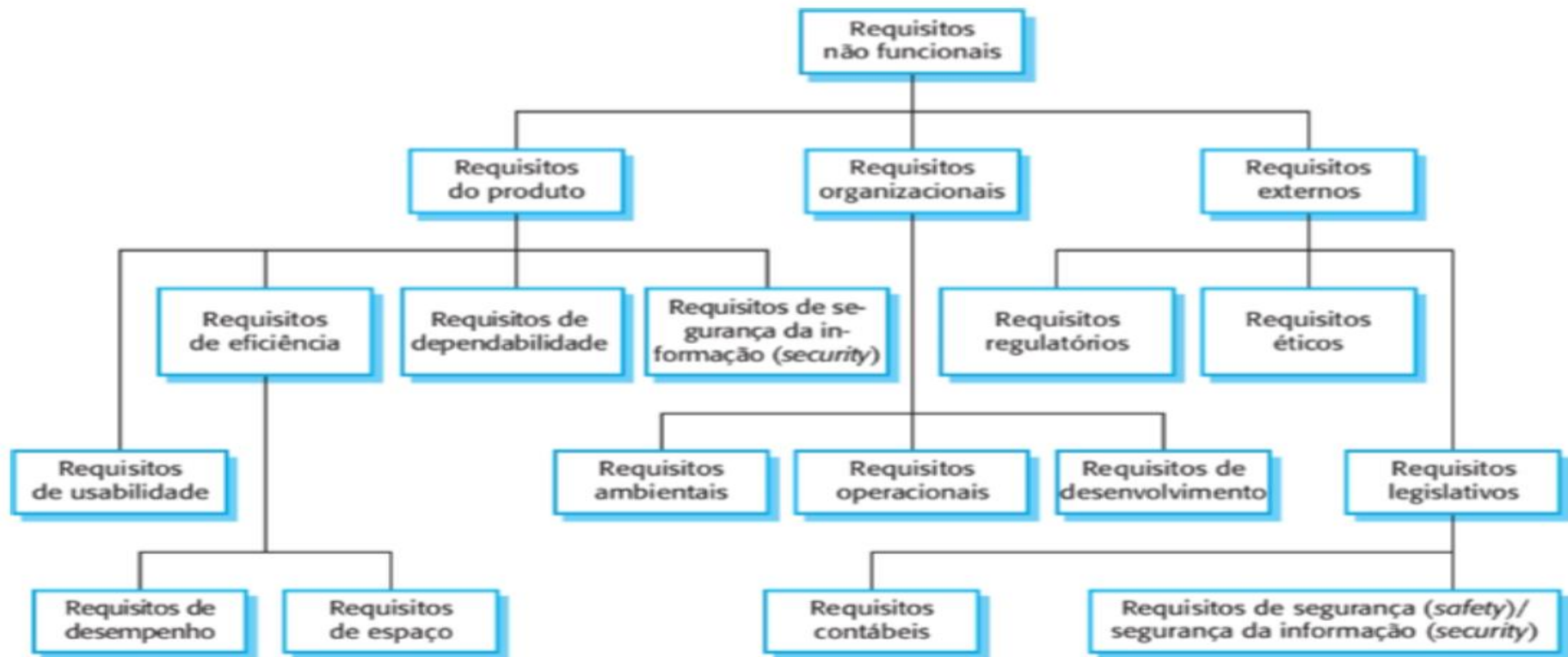


# Classificação dos Requisitos Não Funcionais



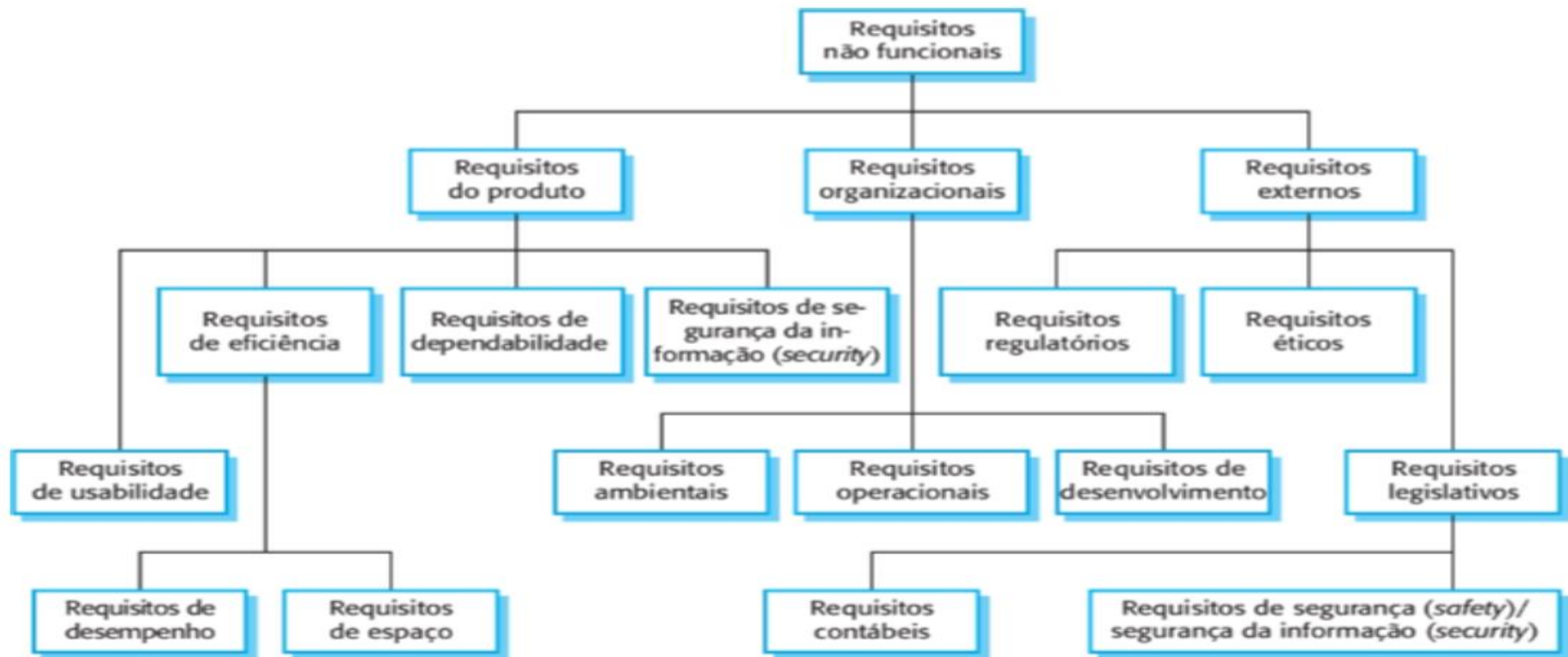
# Classificação dos Requisitos Não Funcionais (cont.)

Requisitos do Produto: Esses requisitos especificam ou restringem o comportamento do software durante a execução. Os exemplos incluem requisitos de desempenho, relativos à rapidez com que o sistema deve executar e de quanta memória ele precisa; requisitos de confiabilidade, que estabelecem a taxa de falha aceitável; requisitos de segurança da informação (*security*): e requisitos de usabilidade.



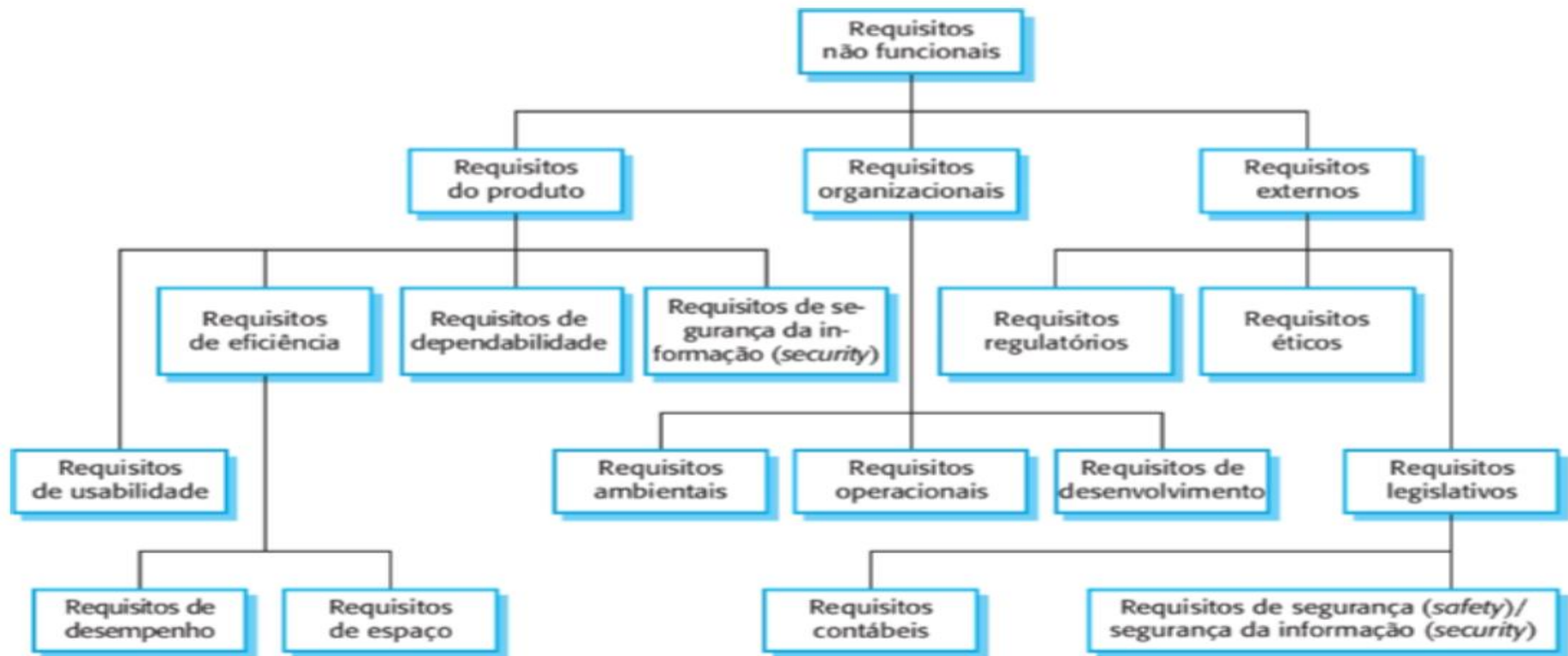
# Classificação dos Requisitos Não Funcionais (cont.)

Requisitos Organizacionais: São requisitos de sistema amplos, derivados das políticas e procedimentos nas organizações do cliente e do desenvolvedor. Os exemplos incluem requisitos de processos operacionais, que definem como o sistema será utilizado; requisitos de processos de desenvolvimento, que especificam a linguagem de programação, o ambiente de desenvolvimento ou os padrões de processo a serem utilizados; e os requisitos ambientais, que especificam o ambiente operacional do sistema.



# Classificação dos Requisitos Não Funcionais (cont.)

Requisitos Externos: são todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento. Podem incluir requisitos regulatórios, que estabelecem o que deve ser feito para o sistema ser aprovado por uma entidade reguladora, como uma autoridade de segurança nuclear; requisitos legislativos, que devem ser seguidos para garantir que o sistema opere dentro da lei; e requisitos éticos, que garantem que o sistema será aceitável para os usuários e para o público em geral.



# Requisitos Não Funcionais (cont.)

- Exemplos de Requisitos do Produto, Requisitos Organizacionais e Requisitos Externos que podem estar incluídos na especificação do Sistema Mentcare:

## **Requisito do produto**

O sistema Mentcare deve ficar disponível para todas as clínicas durante o expediente normal (segunda-sexta, 8h30-17h30).

O tempo que o sistema pode permanecer fora do ar no expediente normal não deve ultrapassar 5 segundos em qualquer dia.

## **Requisito organizacional**

Os usuários do sistema Mentcare devem se identificar usando o cartão de identificação de autoridade de saúde.

## **Requisito externo**

O sistema deve implementar providências para a privacidade do paciente, conforme estabelecido em HStan-03-2006-priv.

# Requisitos Não Funcionais (cont.)

- Um problema comum com os requisitos não funcionais é que os *stakeholders* geralmente propõem esses requisitos na forma de metas gerais, como a facilidade de uso, a capacidade do sistema para se recuperar de uma falha ou a resposta rápida do usuário.
  - As metas estabelecem boas intenções, mas causam problemas para os desenvolvedores do sistema, uma vez que abrem espaço para interpretação e subsequente conflito após o sistema ser entregue.
  - Eles geram metas não testáveis.

# Requisitos Não Funcionais (cont.)

- Exemplo: a meta do Sistema Mentcare a seguir é um exemplo típico de como os requisitos de usabilidade seriam solicitados por um gestor:

*O sistema deve ser fácil de usar pela equipe médica e ser organizado de tal modo que os erros de usuário sejam minimizados.*

- Exemplo: a meta do Sistema Mentcare a seguir é um exemplo de meta verificável (testável).

*A equipe médica deve ser capaz de utilizar todas as funções do sistema após duas horas de treinamento. Após essa etapa, a quantidade média de erros cometidos pelos usuários experientes não deve ultrapassar dois erros por hora de uso do sistema.*



# Requisitos Não Funcionais (cont.)

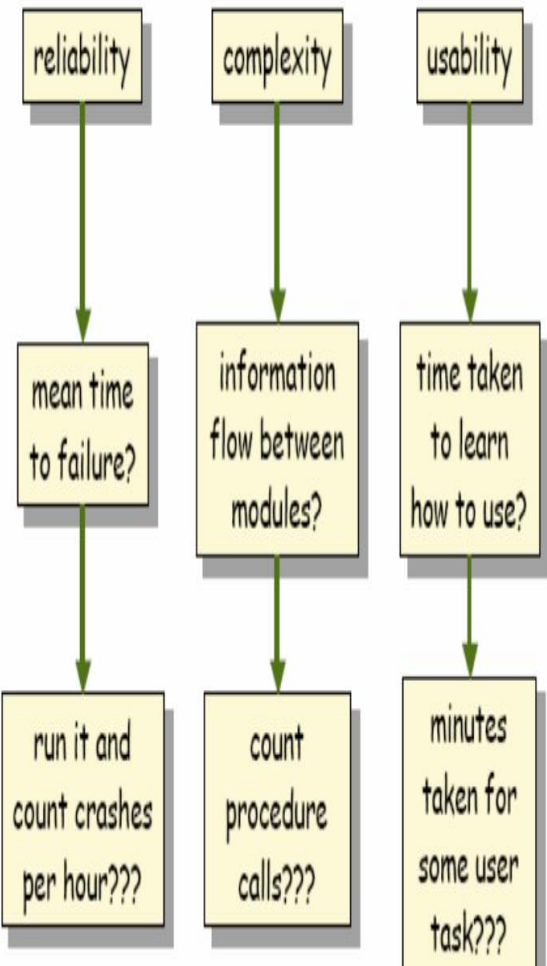
- Sempre que possível, requisitos não funcionais devem ser escritos de forma quantitativa para que possam ser testados objetivamente.
  - A figura na próxima transparência exibe as métricas para especificar as propriedades não funcionais do sistema.
  - É possível mensurar essas características quando o sistema estiver sendo testado, para conferir se ele cumpriu ou não seus requisitos não funcionais.

# Métricas para Requisitos Não Funcionais

Propriedade	Métrica
Velocidade	Transações processadas/segundo
	Tempo de resposta do usuário/evento
	Tempo de atualização da tela
Tamanho	Megabytes/número de chips de ROM
Facilidade de uso	Tempo de treinamento
	Número de quadros de ajuda
Confiabilidade	Tempo médio até a falha
	Probabilidade de indisponibilidade
	Taxa de ocorrência de falhas
	Disponibilidade
Robustez	Tempo para reiniciar após a falha
	Porcentagem de eventos causando falhas
	Probabilidade de corromper dados em uma falha
Portabilidade	Porcentagem de declarações dependentes do sistema-alvo
	Número de sistemas-alvo

# Requisitos Não Funcionais (cont.)

examples...



- Na prática, os clientes de um sistema costumam achar difícil traduzir suas metas para requisitos mensuráveis.

- Para algumas metas, como manutenibilidade, não há métricas simples que possam ser utilizadas.
- Em outros casos, quando é possível fazer uma especificação quantitativa, os clientes podem não conseguir relacionar suas necessidades com essas especificações.
- Eles não entendem o que algum número definindo confiabilidade significa em termos de experiência cotidiana com sistemas computadorizados.
- O custo de verificar objetivamente os requisitos não funcionais mensuráveis pode ser alto e os clientes que pagam pelo sistema podem achar esses valores não justificáveis.

# Requisitos Não Funcionais (cont.)

- Frequentemente, os requisitos não funcionais entram em conflito e interagem com outros requisitos funcionais ou não funcionais.
- Exemplo:
  - O requisito de identificação na figura abaixo requer que uma leitora de cartão instalada em cada computador conectado ao sistema.
  - Pode haver outro requisito que exija acesso móvel ao sistema, por meio de *tablets* e *smartphones* dos usuários.
  - Esses dispositivos normalmente não são equipados com leitoras de cartão e portanto, nessas circunstâncias, pode ser necessário o suporte para algum método de identificação alternativo.

## Requisito organizacional

Os usuários do sistema Mentcare devem se identificar usando o cartão de identificação de autoridade de saúde.

# Requisitos Não Funcionais (cont.)

- Pode ser difícil separar os requisitos funcionais dos requisitos não funcionais no documento de requisitos.
  - Se os requisitos não funcionais forem declarados separadamente dos funcionais, a relação entre eles pode ser difícil de compreender.
  - Em condições ideais, deve-se destacar os requisitos claramente relacionados às propriedades emergentes do sistema, como desempenho ou confiabilidade.
  - É possível fazer isso colocando-os em uma seção separada do documento de requisitos ou distinguindo-os, de alguma forma, dos demais requisitos de sistema.

