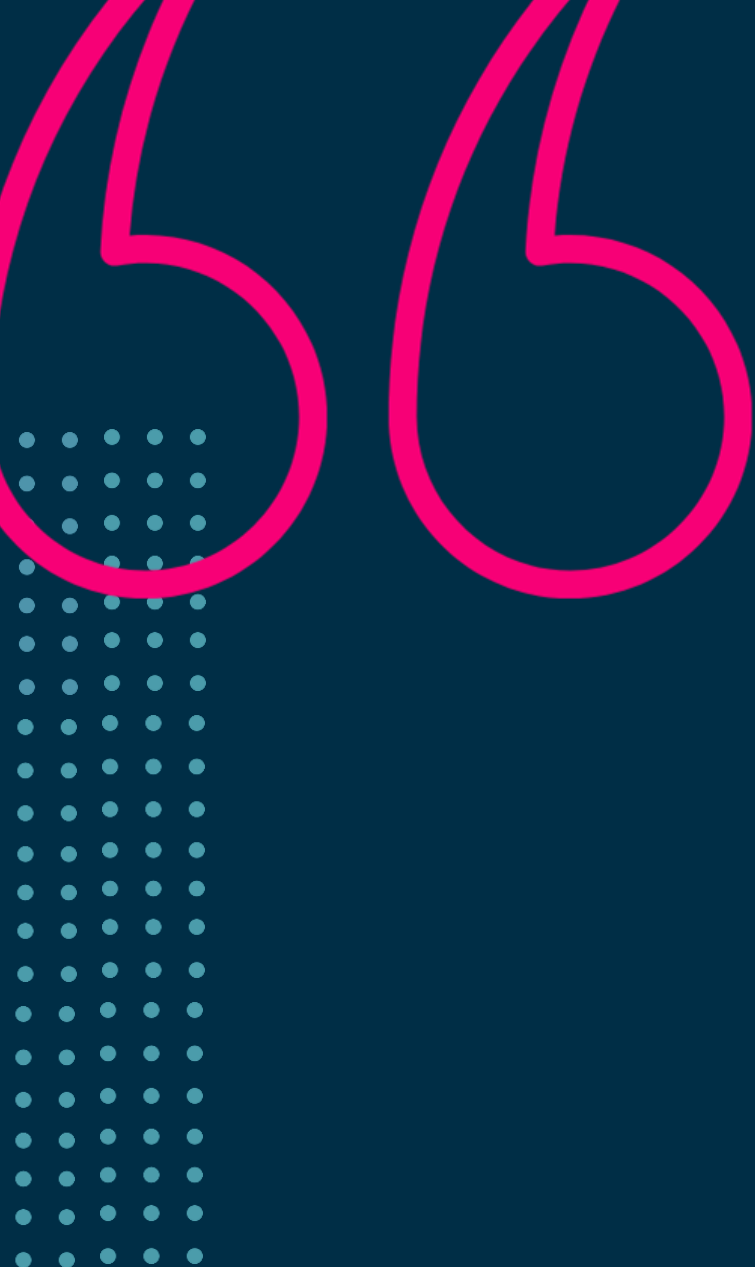


# Engenharia de Requisitos





**“A partir da entrega, o requisito  
passa a viver para sempre dentro  
do produto, até que alguma  
solicitação de mudança o elimine  
ou modifique”**

**Sheila Reineh**

# Aula 08

## Unidade V - Gerenciamento de Requisitos de Software

- **Evolução dos Requisitos de Software**

## Evolução dos Requisitos

- Os sistemas geralmente **REFLETEM SITUAÇÕES DO MUNDO REAL** e, com isso, há uma necessidade que o software mude acompanhando as **MUDANÇAS DE REQUISITOS** impostos pelo ambiente em que está inserido.
- Se o sistema não sofre essas mudanças, pode ficar obsoleto e cair em desuso.



# Evolução dos Requisitos

- O **ENVELHECIMENTO DE UM SOFTWARE** é um processo inevitável.

Tem dois tipos de envelhecimento:

- 1º Caso: mudanças necessárias não são implementadas e o sistema não é adequado às novas regras de negócio utilizadas.
- 2º Caso: adaptações são feitas de maneira desordenada e acarretam problemas para o sistema como um todo, gerando novos erros e diminuindo sua manutenibilidade.

## Evolução dos Requisitos

**Exemplo do 1º Caso:** um software que no passado funcionava perfeitamente, mas devido ao fato de seu sistema operacional ter caído em desuso e não existir uma nova versão do software para um sistema operacional mais recente foi esquecido por seus usuários.

**Exemplo do 2º Caso:** ocorre quando mudanças no software são feitas por desenvolvedores que não entendem a estrutura original deste, o que fatalmente implicará em algum dano para esta estrutura, que mesmo pequeno irá aumentando conforme novas atualizações forem sendo feitas, e com o passar do tempo cada nova mudança se tornará mais difícil e conseqüentemente cara. Caso não seja feita uma reestruturação do software, este chegará em um ponto onde novas atualizações ficarão inviáveis.

# Evolução dos Requisitos

As **DESVANTAGENS** causadas pelo envelhecimento de um software são:

- A perda de desempenho devido a modificações não adequadas na sua estrutura interna.
- Geração de novos erros devido a alterações indevidas no código.
- Perda de usuários devido à falta de meios para concorrer com versões mais recentes de sistemas semelhantes.

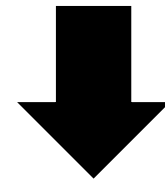


# Evolução dos Requisitos

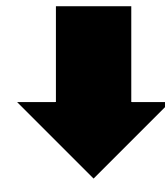
**E o que fazer para evitar esses efeitos?**

## Evolução dos Requisitos

**E o que fazer para evitar esses efeitos?**



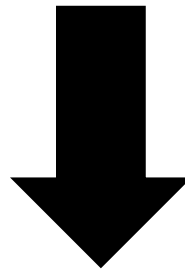
Apesar de inevitável, estes efeitos podem ser retardados ou consideravelmente diminuídos, desde que sejam seguidos alguns cuidados no **DESENVOLVIMENTO E EVOLUÇÃO DO SOFTWARE** em questão.



a **EVOLUÇÃO DE SOFTWARE** compreende as mudanças que irão ocorrer em um programa a fim de deixá-lo completo e, se possível, livre de erros.

## Evolução dos Requisitos

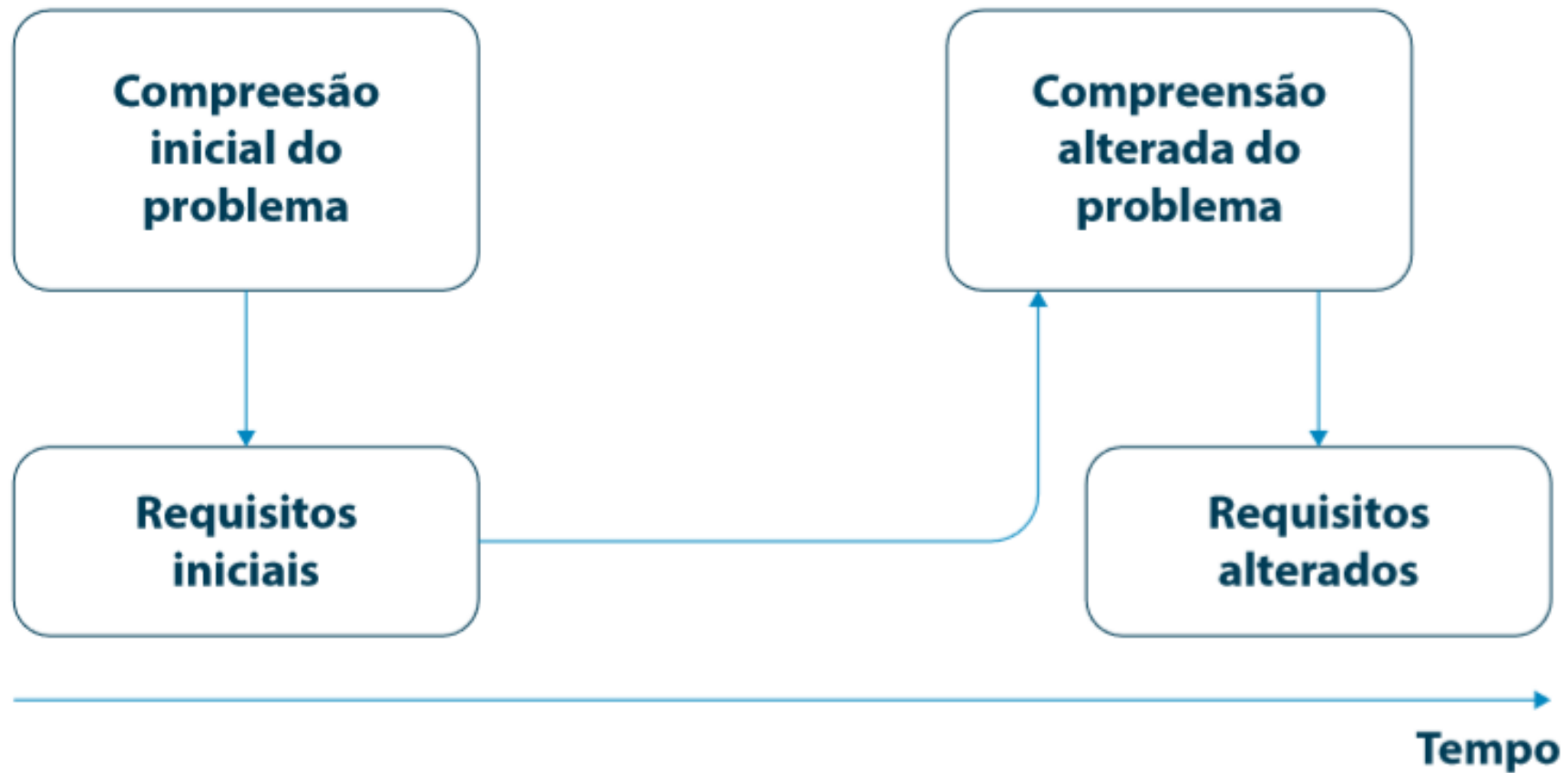
- O **GERENCIAMENTO DE REQUISITOS** visa tratar as **SOLICITAÇÕES DE MUDANÇAS** que irão inevitavelmente acontecer ao longo da vida de um produto de software, provendo mecanismos de **rastreabilidade** que apoiam a tomada de decisão



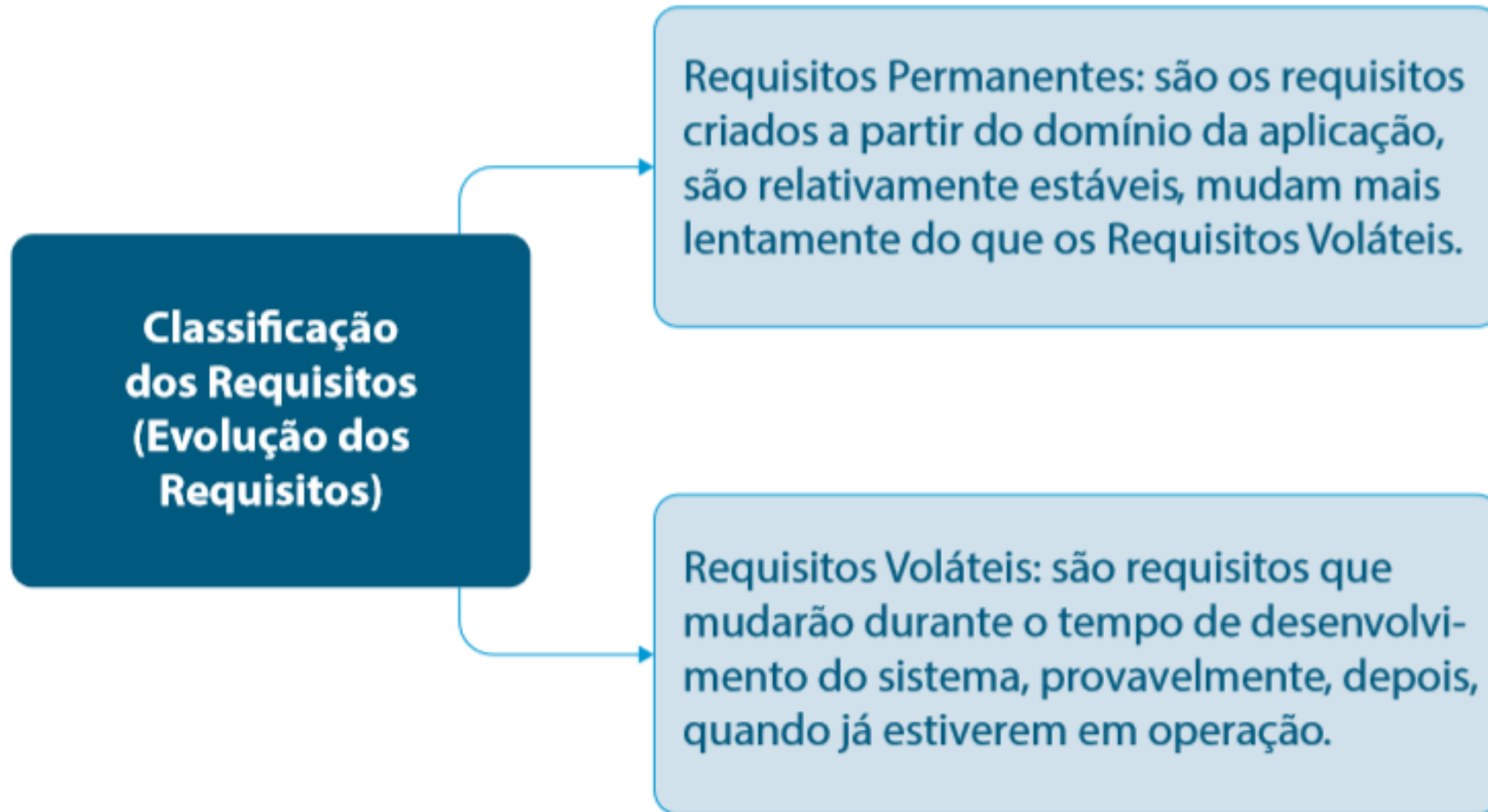
Os requisitos para sistemas de software estão sempre mudando. Porque o problema não pode ser totalmente definido, os requisitos de software são obrigados a ser incompletos

# Evolução dos Requisitos

## Processo de evolução dos requisitos



# Evolução dos Requisitos



# Evolução dos Requisitos - exemplos

## Requisitos Permanentes:

### ■ Sistema de Recursos Humanos (SRC):

Em sistemas deste tipo, sempre existirão os requisitos relacionados à documentação dos funcionários/ colaboradores ou requisitos que fazem referências à empresa ou requisitos referentes ao calendário básico de vencimentos.

### ■ Sistema Hospital:

Em sistemas deste tipo, sempre haverá médicos, enfermeiros e atendentes, ou seja, é padrão disponibilizarem uma aba/janela/ menu referente aos cadastros básicos.

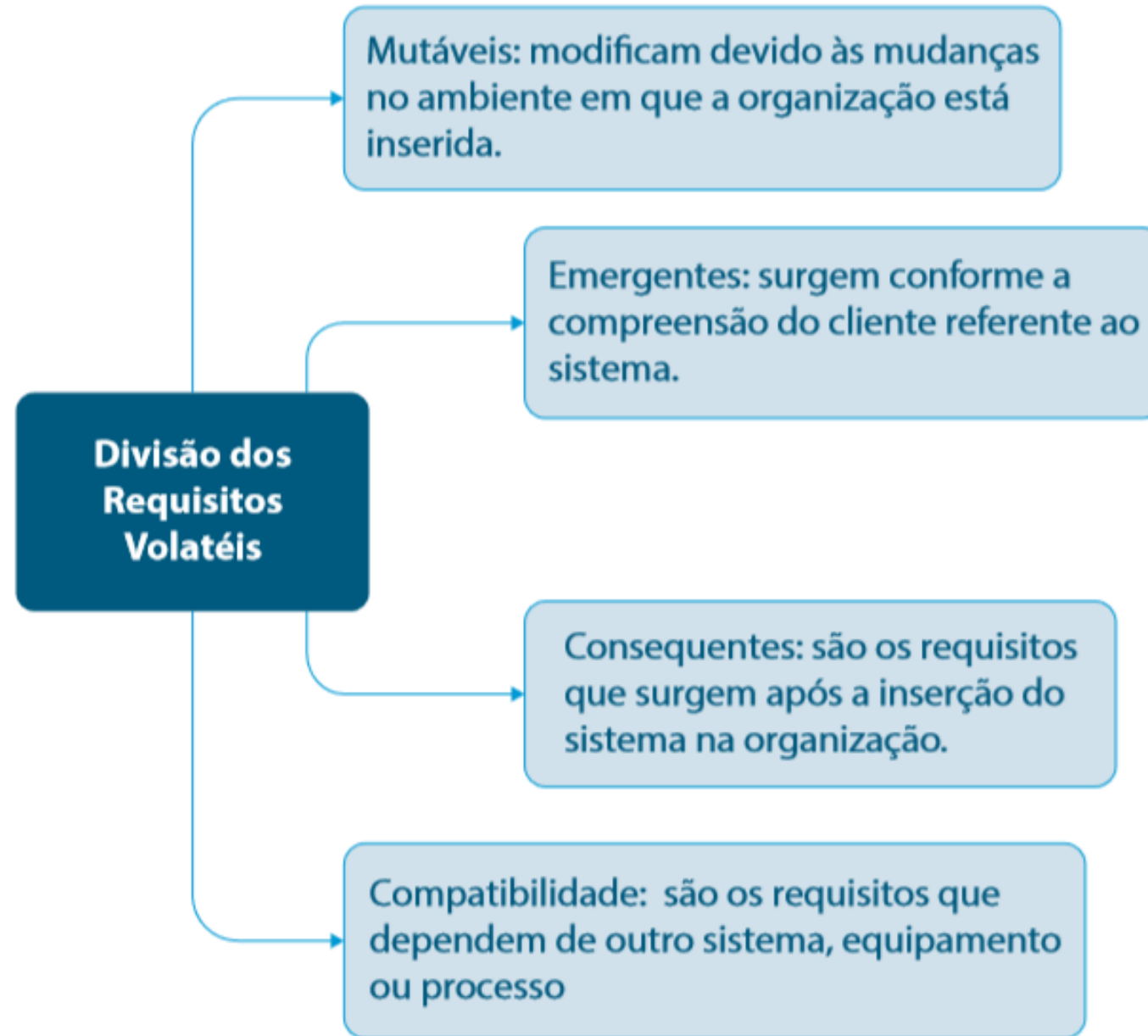


# Evolução dos Requisitos - exemplos

## Requisitos Voláteis:

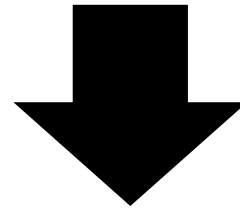
- **Sistema de Recursos Humanos (SRC):** as regras de pagamento dos funcionários/colaboradores podem sofrer alterações por imposição de leis ou por mudança na política da empresa.
- **Sistema Hospital:** um sistema de hospital tem requisitos derivados da política de saúde.
- **Sistema de Vendas:** um sistema que emite notas fiscais tem requisitos derivados das políticas tributárias.

# Evolução dos Requisitos



# Evolução dos Requisitos

Mas o que torna o **GERENCIAMENTO DOS REQUISITOS** complexo?



A mudança em um requisito costuma gerar fortes impactos em outros requisitos já implementados ou não no sistema e isso é assustador.

**E O QUE FAZER?**

# Evolução dos Requisitos

## 1. Estruturar o software para a evolução

- Deve-se fazer sua estrutura visando facilitar a evolução.
- Como não é possível saber quais mudanças serão feitas no futuro, deve-se, portanto avaliar as partes do software que estarão mais sujeitas a mudanças no decorrer de sua vida útil e desenvolvê-las de forma que estas mudanças ocorram mais facilmente.

# Evolução dos Requisitos

## 2. Documentar adequadamente

- Documentação de um projeto deve ser escrita por pessoas qualificadas e sempre que necessário que seja atualizada a contento à medida que novas mudanças forem sendo feitas no código.
- A documentação poderá ser usada para atualizar o sistema daqui a vários anos, e é bem possível que estas atualizações não sejam feitas pelos autores originais, logo deve-se fazer um esforço adicional para que a documentação seja **clara e concisa** com o software, e que seja compreendida por outros que não os seus autores.

# Evolução dos Requisitos

## 3. Revisar a estrutura

- Deve-se ter em mente que sempre que a estimativa de vida útil do software seja longa, revisões da estrutura são fundamentais.
- Caso exista equipe própria para a manutenção, é sempre uma boa política deixá-la participar da revisão.
- Revisões devem começar antes mesmo da codificação, tais revisões são baratas, rápidas e podem poupar muito tempo e recursos no futuro. (estranhamente estas revisões são muito pouco utilizadas na prática).



