



## Unidade 4 – Princípios de Engenharia de Requisitos Parte 1

### CERTIFICAÇÃO CPRE-FL



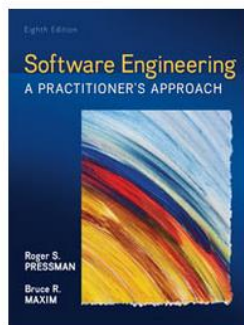
Prof. Aparecido V. de Freitas  
Doutor em Engenharia  
da Computação pela EPUVSP

[aparecidovfreitas@gmail.com](mailto:aparecidovfreitas@gmail.com)

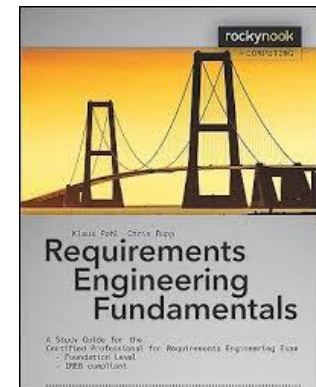
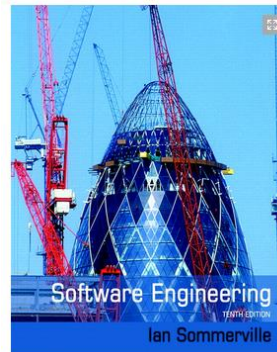


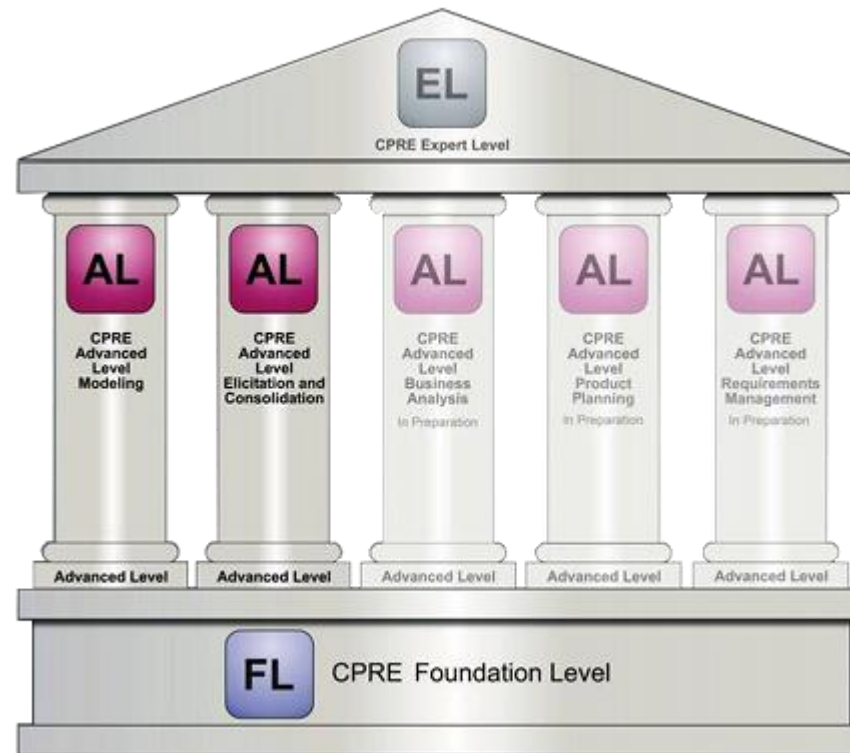
# Bibliografia

- **Fundamentos de Engenharia de Requisitos – POHL K., RUPP C. – IREB 2012**
- **Software Engineering – A Practitioner's Approach – Roger S. Pressman – Eight Edition – 2014**
- **Software Engineering – Ian Sommerville – 10<sup>th</sup> edition - 2015**
- Engenharia de Software – Uma abordagem profissional – Roger Pressman - McGraw Hill, Sétima Edição - 2011
- Engenharia de Software – Ian Sommerville – Nona Edição – Addison Wesley, 2007
- Requirements Engineering Fundamentals – IREB Compliant – Klaus Pohl e Chris Rupp – **CPRE-FL**, 1996.



Software Engineering: A Practitioner's Approach, 8/e





**CPRE-FL** (Certified Professional Requirements Engineer – Foundation Level)



## Certificação em Engenharia de Requisitos

- **IREB – International Requirements Engineering Board** - entidade criada em 2006 com a visão de criar uma base internacionalmente aceita de profissionalização da disciplina Engenharia de Requisitos;
- ABRAMTI – Associação Brasileira de Melhoria em TI;
- IBTQS – Instituto Brasileiro de Qualidade em Testes de Software;
- Certificação CPRE – Certified Professional for Requirements Engineering;
  - ✓ Nível Fundamental
  - ✓ Nível Avançado
  - ✓ Nível Especialista





## Certificação em Engenharia de Requisitos – CPRE

- Sem data de expiração;
- Validade Internacional;
- Quatro provas por ano em locais definidos pela ABRAMTI em todo o território nacional;
- Não há pré-requisitos para a realização da prova;
- Não há necessidade de comprovar experiência na área de Engenharia de Requisitos;





## Certificação em Engenharia de Requisitos – CPRE

- A prova CPRE é composta por **45** questões com duração de **60** minutos;
- A pontuação das questões é variável dependendo do grau de importância do assunto;
- Questões assinaladas incorretamente invalidam acertos de outras questões;
- Há questões com mais de uma alternativa correta;
- O enunciado da questão informa quantas respostas certas devem ser assinaladas;
- A aprovação é concedida para os candidatos com pelo menos **60%** de acertos das questões.





## Competências do Engenheiro de Requisitos

- Estar ciente das razões que podem levar a uma Engenharia de Requisitos inapropriada;
- Compreender os diversos tipos de Requisitos de Software;
- Compreender a importância dos Requisitos de Qualidade;
- Estar ciente da importância da Comunicação da Engenharia de Requisitos;
- Ter proficiência nas atividades principais da Engenharia de Requisitos;
- Conhecer as responsabilidades e competências do Engenheiro de Requisitos.







Em que situações um projeto de software é bem-sucedido ?







Em que situações um projeto de software é bem-sucedido ?





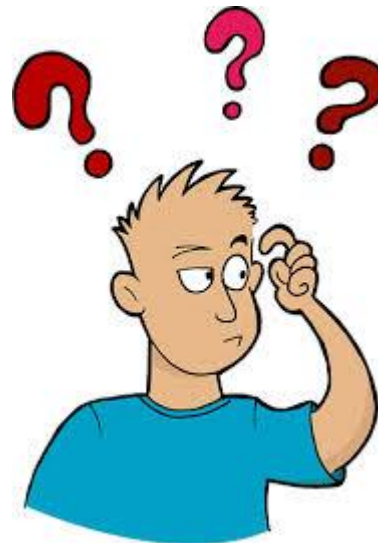
## Em que situações um projeto de software é bem-sucedido ?

- ✓ Prazos foram cumpridos !
- ✓ Custos próximos aos planejados !
- ✓ Software agregou valor ao negócio !
- ✓ Especificações de qualidade foram atendidas !
- ✓ Funcionalidades foram atendidas !
- ✓ Processo de definição de requisitos foi adequado !





Será que todos os projetos de Software são assim ?





## Serã que todos os projetos de Software são assim ?



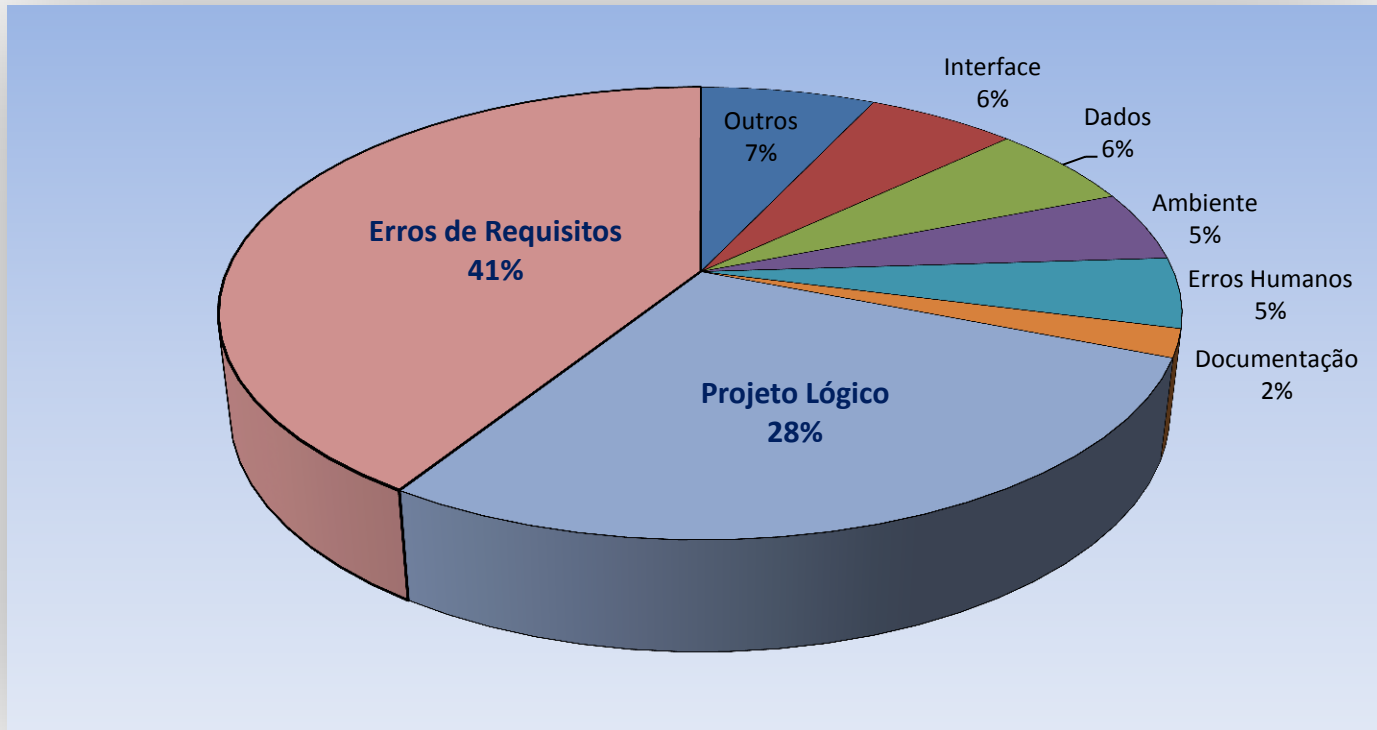
- Estudos mostram que 40% a 60% dos problemas ocorridos em projetos de software, são decorrentes de falhas no processo de requisitos.



Aurum A. ; Wohlin C. – Engineering and Managing Software Requirements



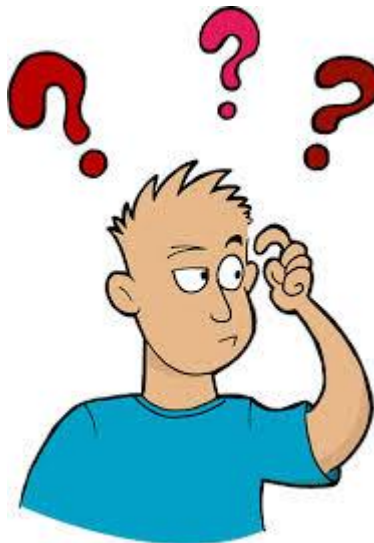
## Erros de Projetos



Fonte: Projeto US Air Force



Quais as consequências de um processo de requisitos mal elaborado ?





## Quais as consequências de um processo de requisitos mal elaborado ?

- Falha da definição de requisitos acarreta defeitos de software;
- Requisitos são fundamentais para a obtenção de Software de qualidade;
- O Software entregue não irá refletir as necessidades reais do stakeholder;
- Alto impacto nos custos e prazos do projeto.







Como desenvolver um bom processo de Requisitos ?





## Como desenvolver um bom processo de Requisitos ?

- Empregar técnicas de elicitação e análise de Requisitos;
- Conhecer o negócio do stakeholder;
- Levantar adequadamente as necessidades do stakeholder;
- Definir requisitos em conjunto com stakeholder (e não isoladamente).



Requirements





*Qual o impacto dos requisitos no custo de um Software ?*





## Impacto dos requisitos no custo de um Software

- Um erro de requisito acarreta:
- Custo 2.5 vezes mais no projeto;
- Custo 5 vezes mais na codificação;
- Custo 10 vezes mais no Teste unitário;
- Custo 25 vezes mais no Teste de aceitação;
- Custo 100 vezes mais na Manutenção

Propagação dos custos



### O alto custo dos Requisitos Errados

A regra do 1-10-100



Custo relativo para reparar erros:  
Quando Introduzidos X Quando reparados.

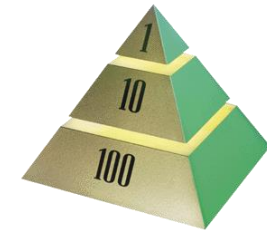
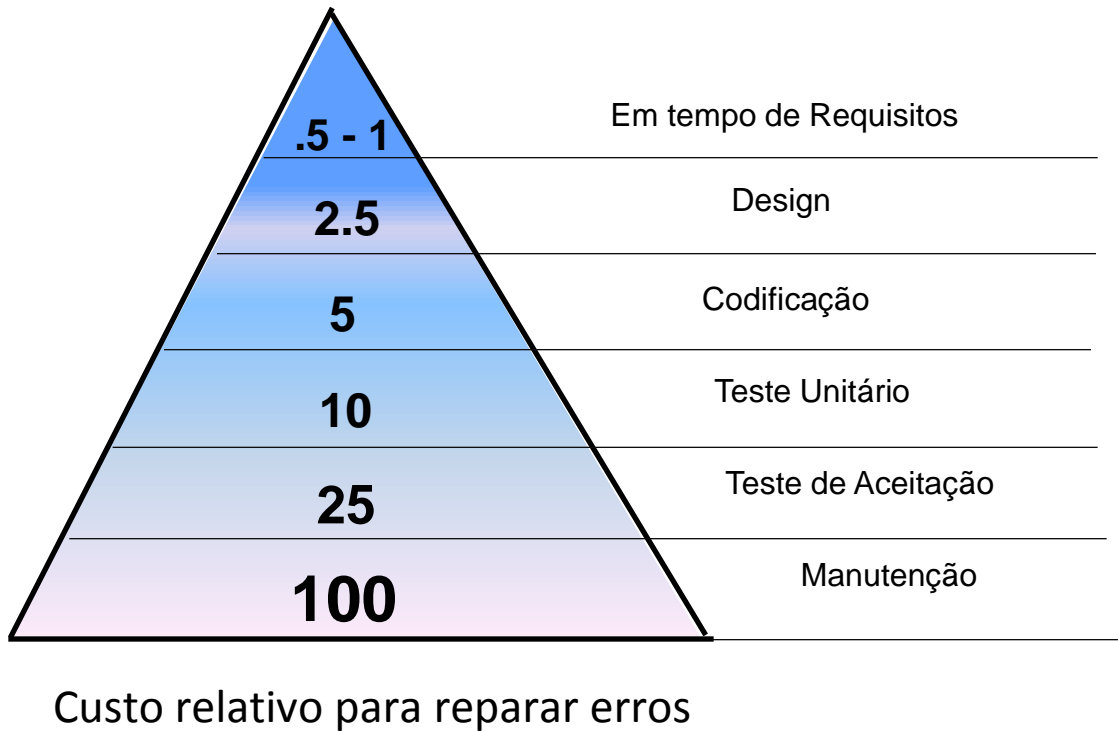
"All together, the results show as much as a 200:1 cost ratio between finding errors in the requirements and maintenance stages of the software lifecycle."

Boehm 1982



## O alto custo dos Requisitos Errados

A regra do 1-10-100



“Os resultados mostram como corrigir erros encontrados nos requisitos custa até 200 vezes menos do que em estágios mais avançados do ciclo de vida”

Fonte: Boehm 1988



Ok, mas então o que é um Requisito ?





# Requisito

- De forma bem abrangente, requisito é uma definição clara, sem ambiguidade e precisa das funcionalidades que o sistema é capaz de realizar, juntamente com restrições, para que os objetivos do negócio do stakeholder sejam atendidos.
- Uma condição ou capacidade que deve ser alcançada ou estar presente em um sistema ou componente de sistema para satisfazer um contrato, norma, especificação ou outro documento formalmente imposto.

**Norma IEEE 610.12-1990 – Página 62**

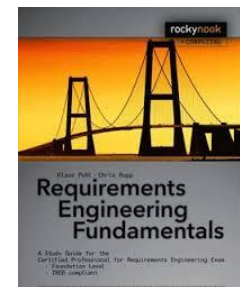






# Engenharia de Requisitos

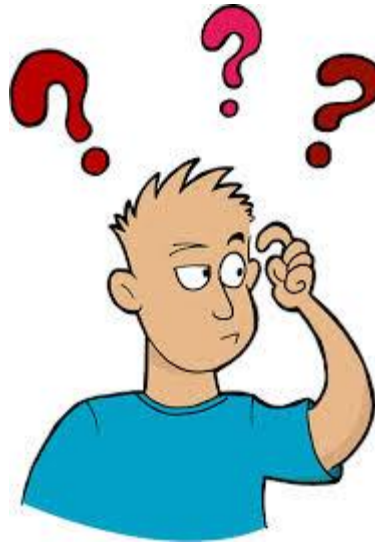
- ✓ Processo de Engenharia que cuida do levantamento, análise, documentação e validação de Requisitos.
- ✓ A Engenharia de Requisitos é uma abordagem sistemática e disciplinada para a especificação e gerenciamento de requisitos.



Fonte: Pohl, 1996



Quais as principais fontes de obtenção dos Requisitos ?

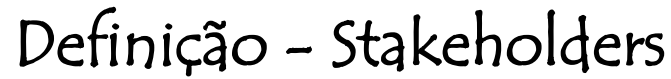




# Stakeholders

- São as mais importantes fontes para obtenção de requisitos;
- São as pessoas ou organizações que têm impacto sobre os requisitos;
- Isso inclui as pessoas que irão interagir com o sistema (por exemplo, usuários e administradores);
- Inclui também pessoas que não irão usar o sistema, mas que têm interesse no sistema (por exemplo, alta gerência ou um cracker do qual o sistema precisa ser protegido...);
- Inclui também entidades legais, instituições, documentação de sistemas legados, etc

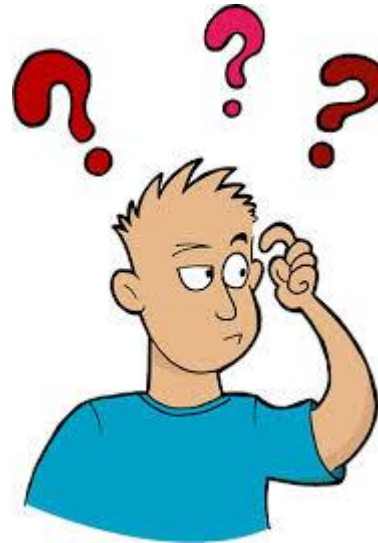




-



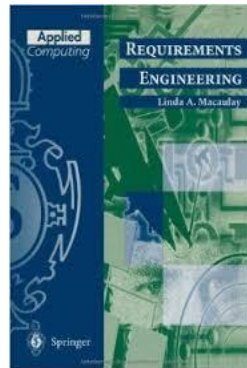
Qual o impacto no desenvolvimento do software quando não se leva em conta todos os Stakeholders ?





## Impacto da consideração dos Stakeholders

- Sem considerá-los, pode-se obter uma **elicitação** fragmentada de requisitos;
- Isso resulta em requisitos incompletos.



Fonte: Macaulay, Requirements Engineering, 1996



Quais os objetivos da Engenharia de Requisitos ?







## Objetivos da Engenharia de Requisitos

- Durante o processo de desenvolvimento, a Engenharia de Requisitos assegura que os requisitos são corretamente elicitados das partes interessadas (stakeholders), que são adequadamente documentados, validados e verificados. Garante também que esses requisitos são gerenciados durante todo o ciclo de vida do software em desenvolvimento.





Quais as atividades centrais da Engenharia de Requisitos ?





## Atividades centrais da Engenharia de Requisitos



■	Concepção	(identificação da necessidade do negócio)
■	Elicitação	(levantamento de requisitos)
■	Análise	(modelagem de requisitos)
■	Negociação	(Tratamento de conflitos e definição de prioridades)
■	Especificação	(Documento textual formal ou modelagem)
■	Validação	(Eliminação de ambiguidade, inconsistências e revisão)
■	Gestão	(Controle de mudanças de requisitos)



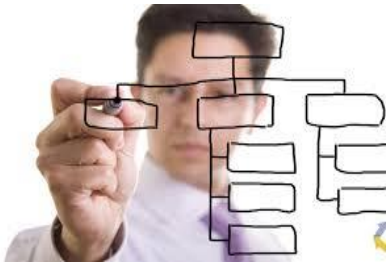
Qual o perfil de um Engenheiro de Requisitos ?





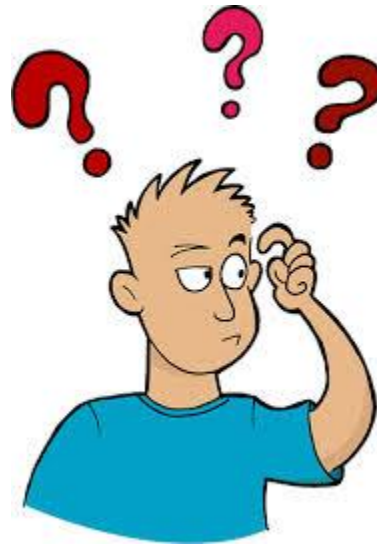
## O Engenheiro de Requisitos

- **Raciocínio Analítico.** Capacidade de familiarizar-se com domínios que lhe sejam desconhecidos.
- **Empatia.** Boa intuição e capacidade de agir como se fosse um Usuário.
- **Competência Comunicativa.** Saber ouvir, fazer perguntas certas na hora correta. Boa capacidade de comunicação.
- **Resolução de Conflitos.** Capacidade de identificar conflitos.
- **Moderação.** Deve ser um moderador nato.
- **Auto-confiança:** Capacidade de se defender em caso de fortes objeções às suas opiniões, jamais levando tais comentários como uma crítica pessoal.
- **Persuasão:** Capacidade de sustentar e defender requisitos dos Usuários em reuniões de equipe e apresentações.





Como a Engenharia de Requisitos é incorporada  
nos modelos de Processo de Software ?





## Engenharia de Requisitos no Modelo Cascata



- No modelo Cascata (Waterfall) [Royce,1987], os requisitos são totalmente elicitados antes do desenvolvimento propriamente dito.
- Nesses processos, a ER é compreendida como uma fase inicial do desenvolvimento de sistemas, finita e restrita no tempo.





## Engenharia de Requisitos no Modelo Evolucionário



- Nos modelos evolucionários elicita-se, inicialmente, somente os requisitos necessários para a implementação, visto que é difícil “prever” funcionalidades futuras;
- Nesse modelo, os requisitos mudam ao longo do projeto e, portanto, a ER é vista como um processo contínuo e abrangente que engloba e integra todas as fases do desenvolvimento do software;
- No início de cada iteração, novos requisitos são identificados e priorizados.



# Importância da Comunicação

- Requisitos precisam ser comunicados;
- Na maioria dos casos, utiliza-se a linguagem natural como forma de comunicação;
- Emissor e receptor usam um código comum intrínseco (por exemplo, a língua portuguesa);
- Quanto mais afinidades existirem entre emissor e receptor, melhor será a troca de informações;
- No entanto, na maioria das vezes, tais condições ideais não existem entre stakeholders;
- Entre o emissor e o receptor, pode haver ruídos na mensagem;
- Isso é amenizado por meio de um glossário, nos quais os termos importantes são explicados;
- Alternativamente, pode-se adotar uma linguagem descritiva formal, por exemplo, a Linguagem de Modelagem Unificada (**UML**).





## Influência do Meio

- Na comunicação verbal, emprega-se redundância (linguagem e gestos, entonação, feedback);
- Na comunicação escrita, pelo contrário, as informações são transmitidas com um mínimo de redundância e feedback.





# Ambiguidade

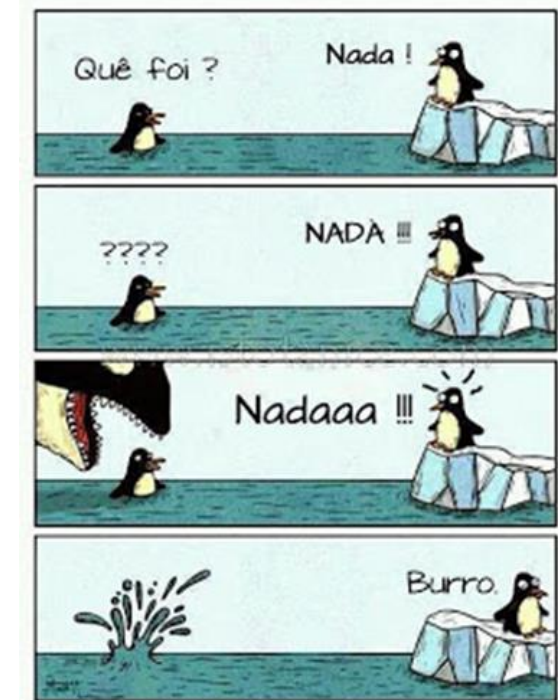
- Muitas vezes, as informações não são adequadamente transmitidas, ou nem mesmo transmitidas;
- O autor pode pressupor que o leitor tenha algum tipo de conhecimento implícito anterior sobre o tema;
- Essas simplificações podem causar problemas em termos de Requisitos, pois os Requisitos podem tornar-se suscetíveis a diversas interpretações (ambiguidade).





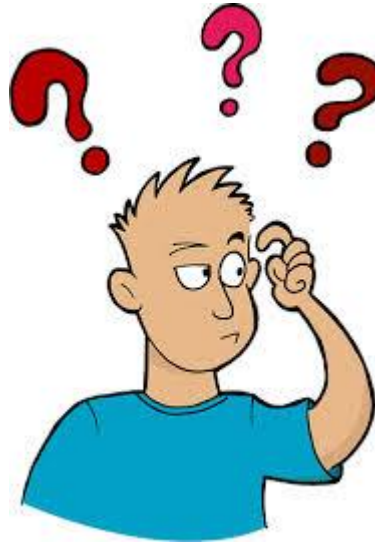
# Ambiguidade

- O velho senhor encontrou o rapaz em seu escritório !
- Visitamos o teatro do povoado que foi fundado em 1980.
- O rapaz viu o mendigo sentado na varanda.
- O policial deteve o bandido que roubou o banco na Av. Brasil.
- O vendedor disse ao cliente que seu preço estava incorreto.





## Quais os tipos de Requisitos ?





# Tipos de Requisitos

- Requisitos funcionais
- Requisitos Não Funcionais ou de Qualidade
- Restrições







# Requisitos Funcionais

- Requisitos funcionais definem a funcionalidade oferecida pelo sistema a ser desenvolvido.
- Um requisito funcional é um requisito relacionado ao resultado de algum comportamento a ser fornecido por uma função do sistema.







# Requisitos Não Funcionais

- Também conhecidos por Requisitos de Qualidade;
- Definem qualidades desejadas do sistema a ser desenvolvido;
- Muitas vezes influenciam a arquitetura do sistema mais que os requisitos funcionais;
- Tipicamente requisitos de qualidade definem o desempenho, a disponibilidade, a confiabilidade, a escalabilidade ou a portabilidade de um sistema.





# Requisitos de Qualidade

- Geralmente requisitos de qualidade costumam ser especificados por meio de linguagem natural;
- Devem ser quantificáveis. Por exemplo, o sistema deverá processar 95% de todas as requisições dentro de 1 segundo. O processamento de todas as requisições não deve ultrapassar 3 segundos em qualquer instante;
- Norma **ISO 9126** auxilia na especificação e categorização dos requisitos de qualidade.



International  
Organization for  
Standardization





## Restrições

- Não podem ser influenciadas pelos membros da equipe;
- Exemplo: O sistema deverá estar disponível no mercado no mais tardar no quarto trimestre de 2016;
- Ao contrário dos requisitos funcionais e de qualidade, as restrições não são implementadas, mas devem ser cumpridas, pois meramente limitam o espaço da solução disponível durante o processo de desenvolvimento.





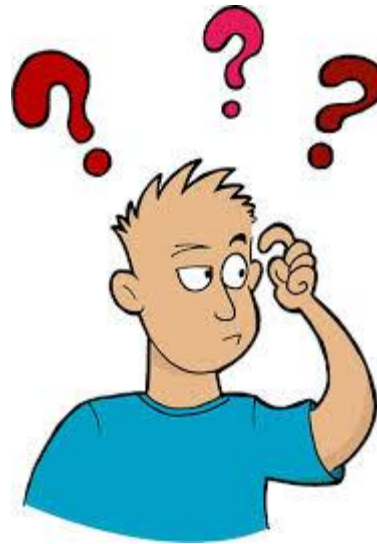
## Os limites do Sistema e do Contexto

- Na Engenharia de Requisitos, o propósito de definir os **limites do sistema** e do **contexto** é identificar a parte do ambiente que influencia os requisitos do sistema a ser desenvolvido.





O que é contexto do sistema ?





## Contexto do Sistema

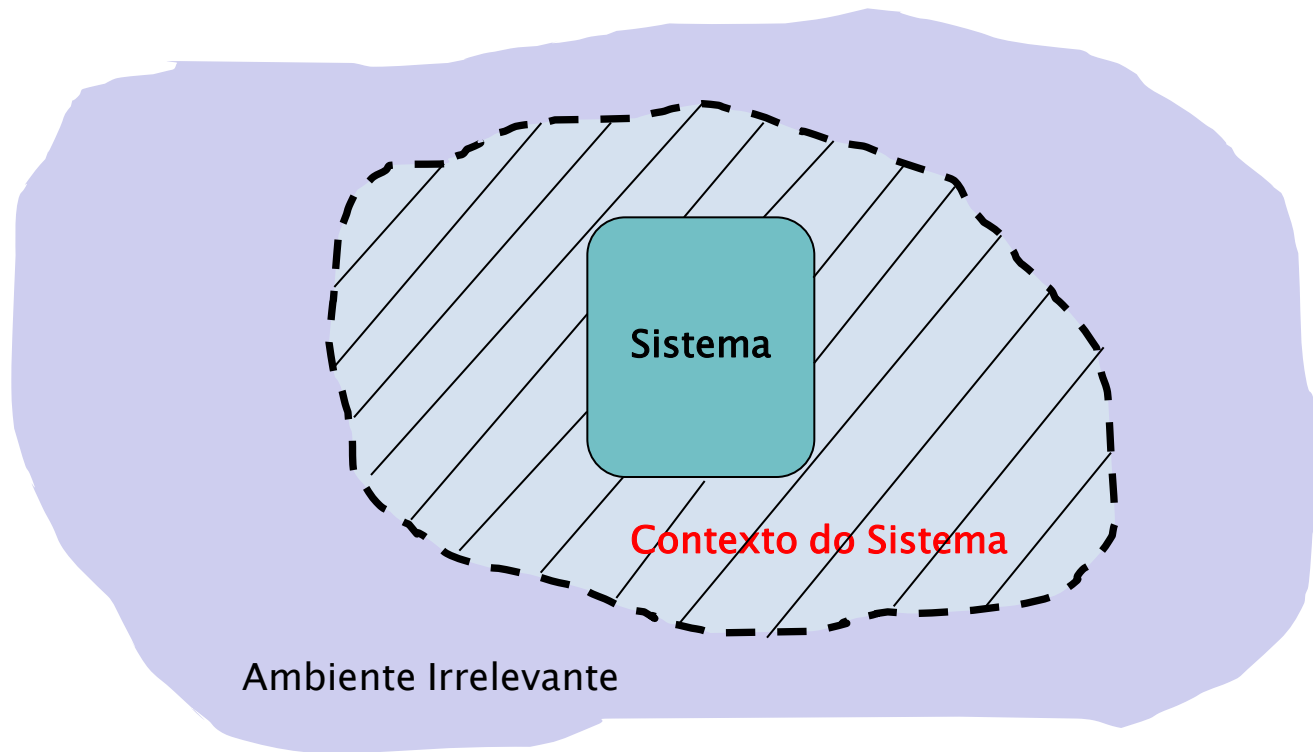
- É a **parte do ambiente** do sistema que é **relevante** para a definição e a compreensão dos requisitos de um sistema a ser desenvolvido;
- Os seguintes aspectos da realidade influenciam o contexto de um sistema:
  - ⊕ Pessoas (usuários ou grupos de usuários)
  - ⊕ Sistemas em operação
  - ⊕ Processos
  - ⊕ Eventos





# Contexto do Sistema

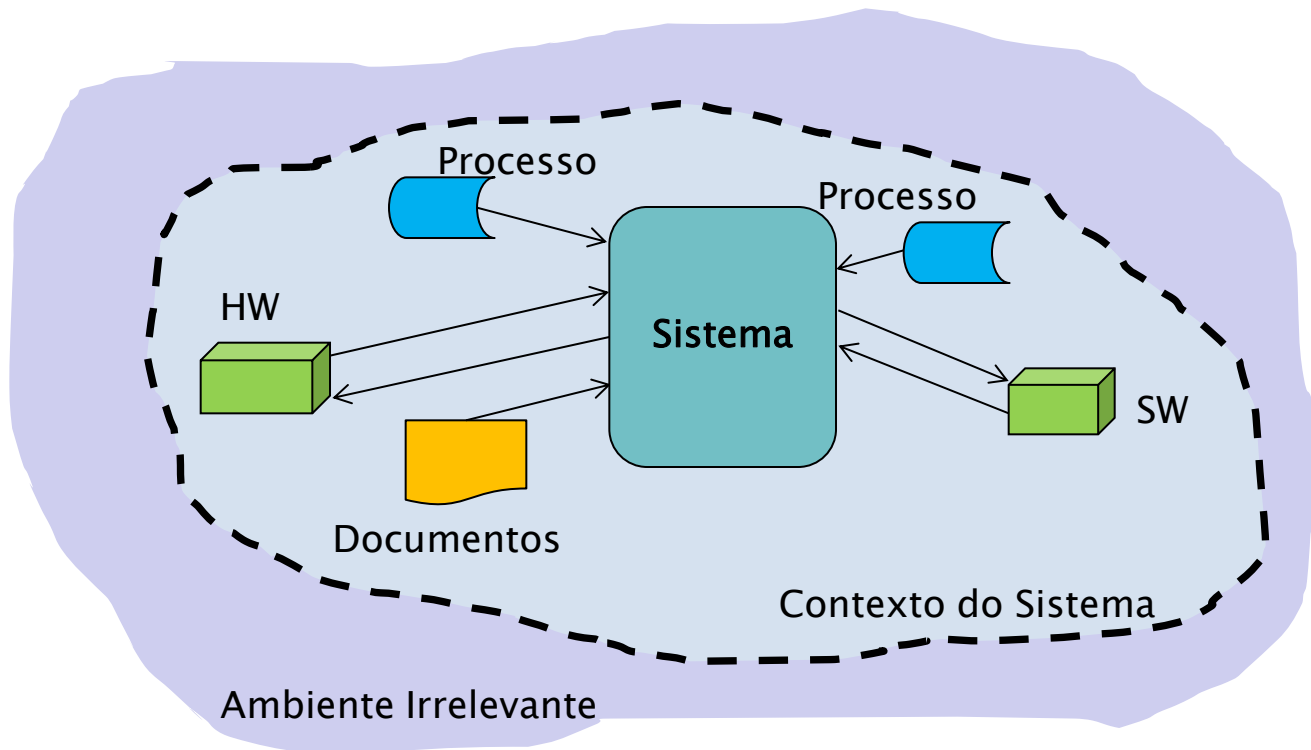
- É responsabilidade do Engenheiro de Requisitos definir adequadamente o contexto do sistema.





# Contexto do Sistema

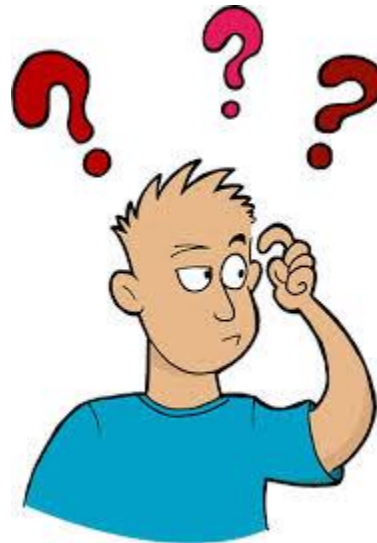
- Pode ser constituído por outros **sistemas**, por grupos de **usuários** que de uma forma ou outra utilizam a interface do sistema a ser desenvolvido, além de fontes adicionais de requisitos e suas interrelações.







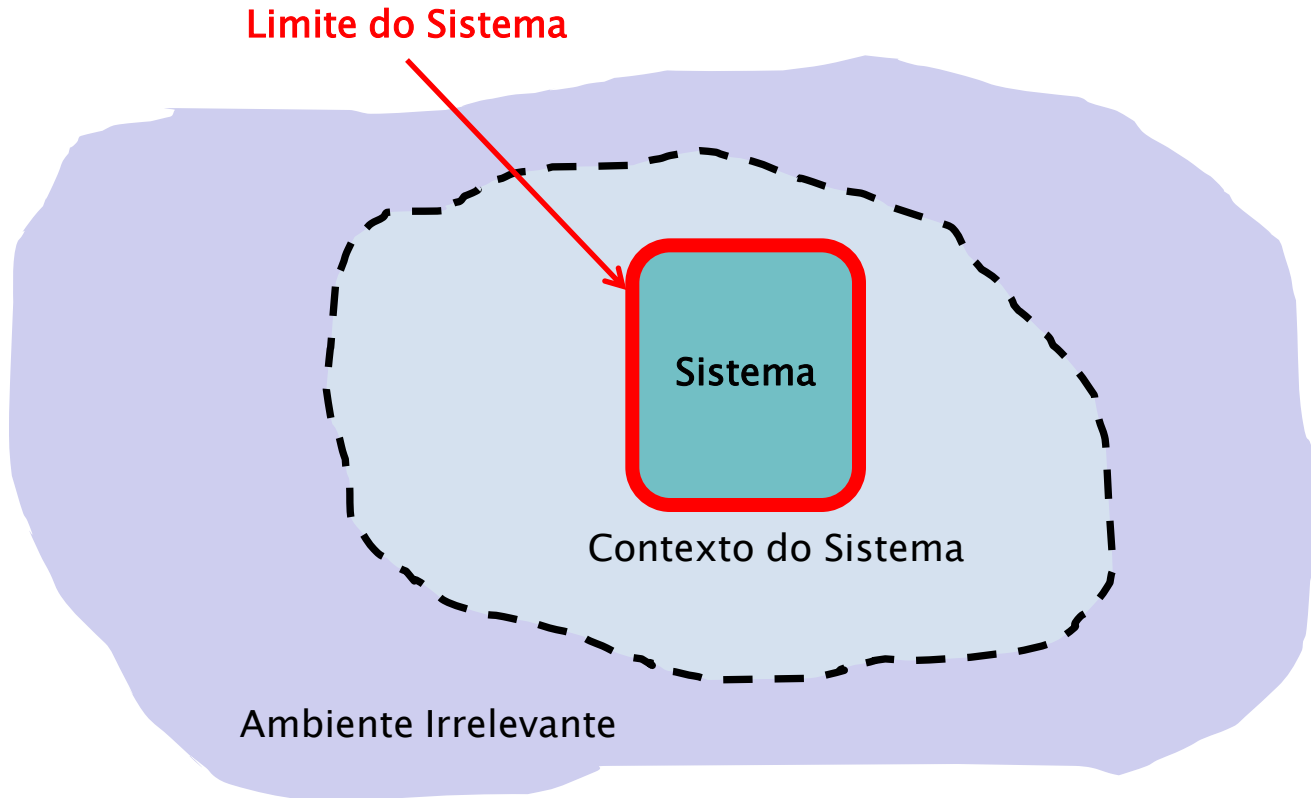
O que é limite do sistema ?





# Limite do Sistema

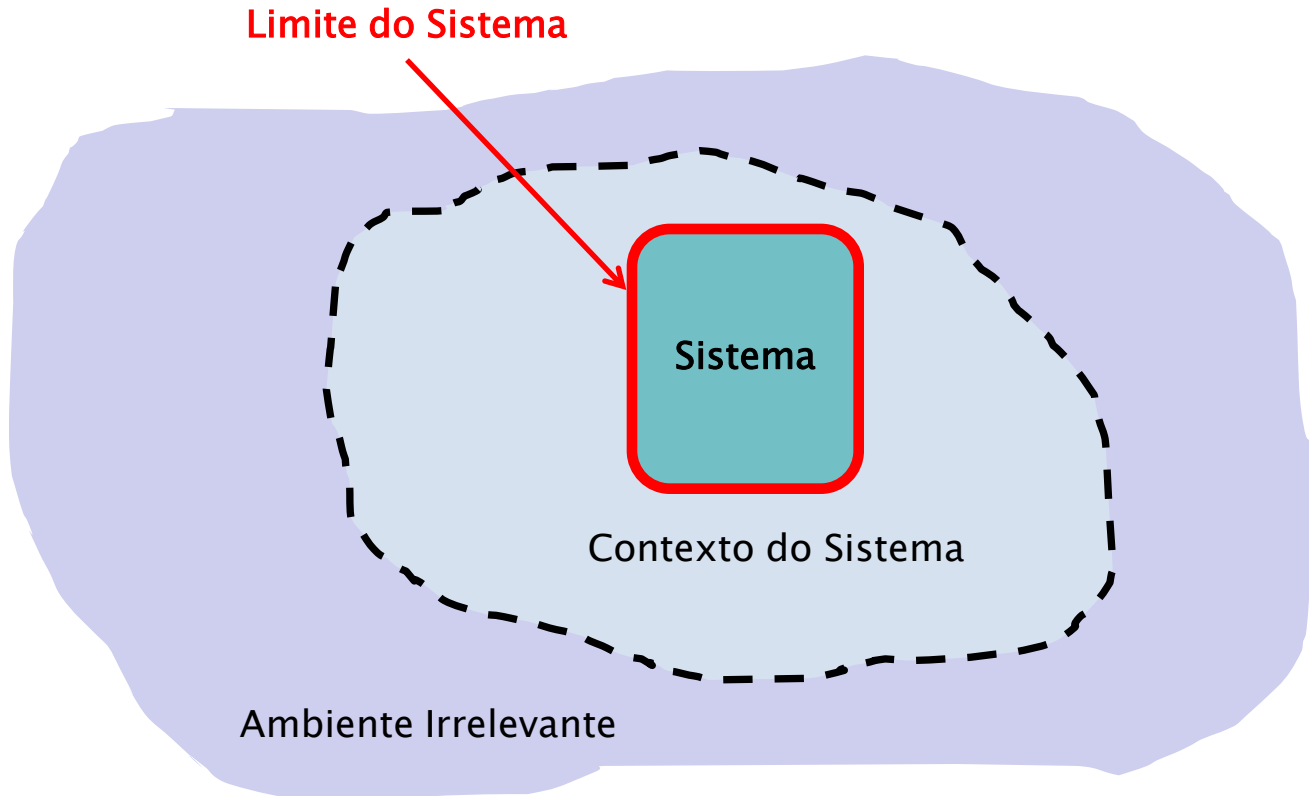
- O **limite do sistema** separa o sistema a ser desenvolvido do seu ambiente (isto é, separa a parte da realidade que pode ser modificada ou alterada pelo processo de desenvolvimento) daqueles aspectos do ambiente que não podem ser mudados ou modificados pelo processo de desenvolvimento.





# Limite do Sistema

- Todos os aspectos que se encontram dentro do **limite do sistema** podem ser alterados durante o desenvolvimento do sistema;
- Com a definição do **Limite do Sistema**, define-se também o **Escopo do Sistema (abrangência do Sistema, fecha-se o escopo do sistema)**.

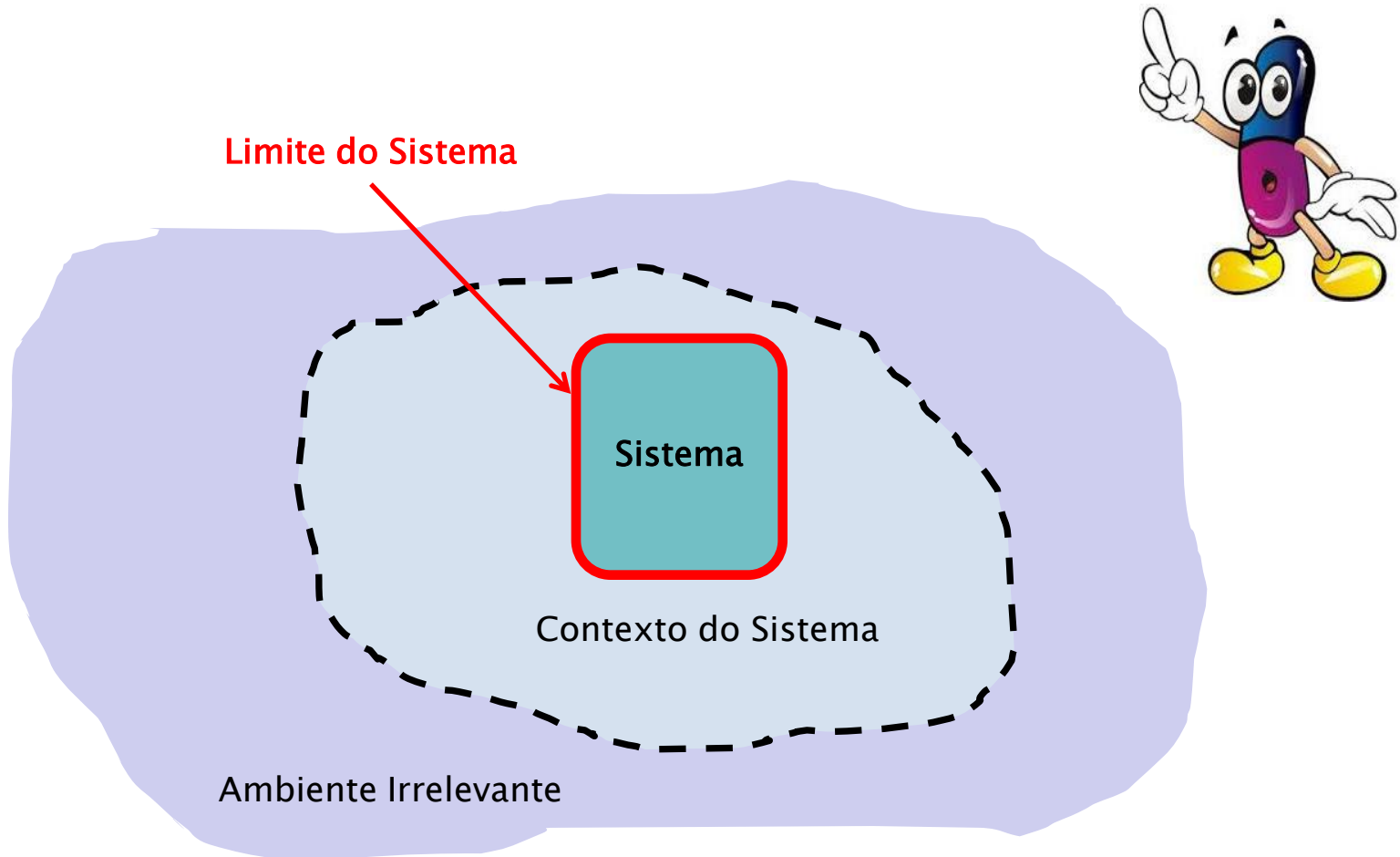




# Limite do Sistema

- Para se definir o **limite do sistema** deve-se responder a seguinte questão:

**Quais aspectos pertencem ao sistema que será desenvolvido ?**

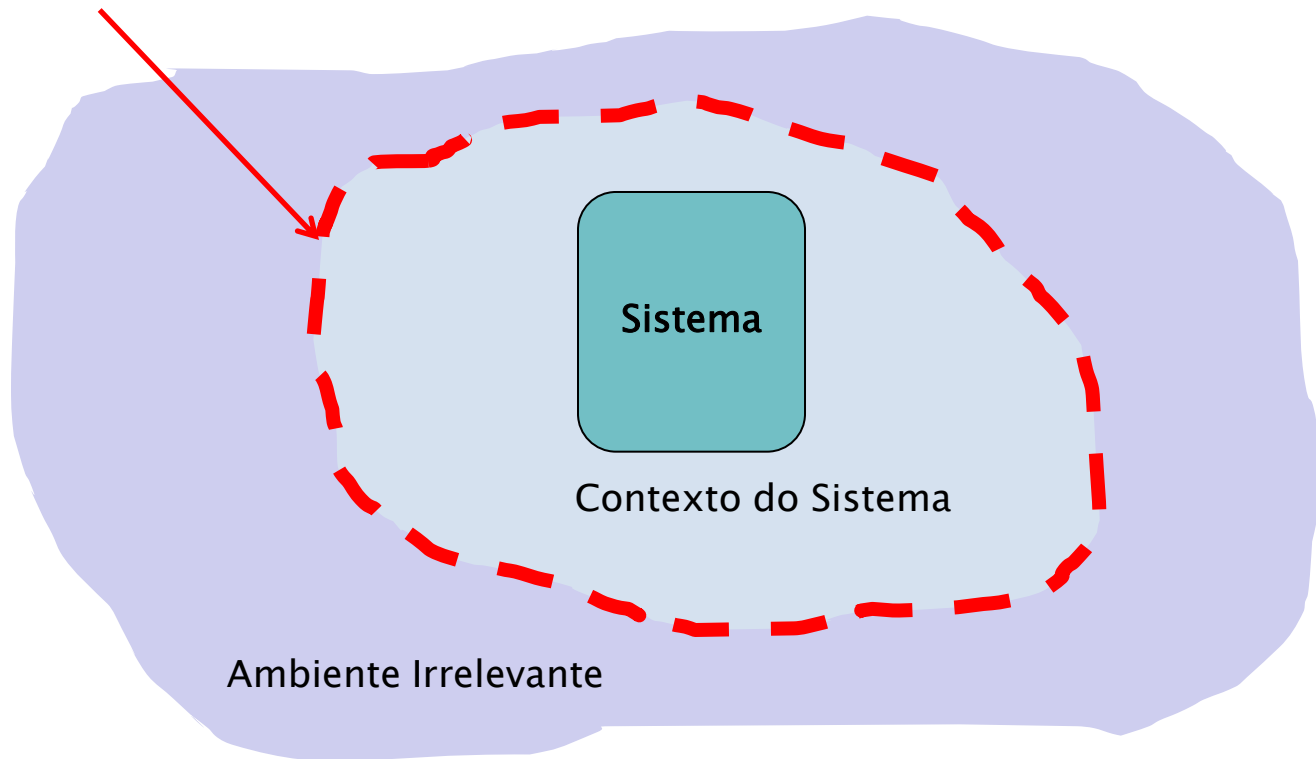




# Limite do Contexto

- O **limite do contexto** separa a parte relevante do ambiente de um sistema a ser desenvolvido da parte irrelevante, isto é, a parte que não influencia o sistema a ser desenvolvido e, sendo assim, não precisa ser considerado durante a Engenharia de Requisitos.

## Limite do Contexto





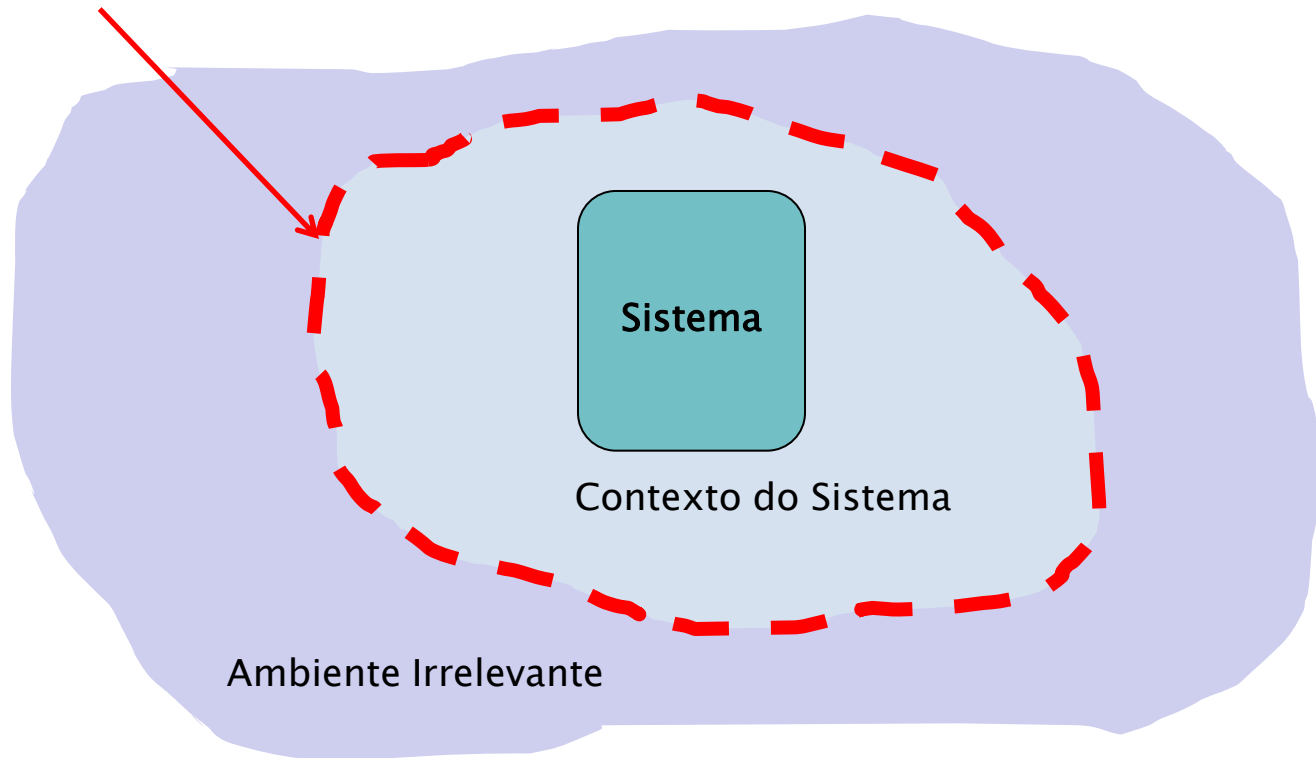
# Limite do Contexto

- Para se definir o Limite do Contexto, deve-se responder a seguinte questão:

**Quais aspectos pertencem ao contexto do sistema e quais são irrelevantes?**



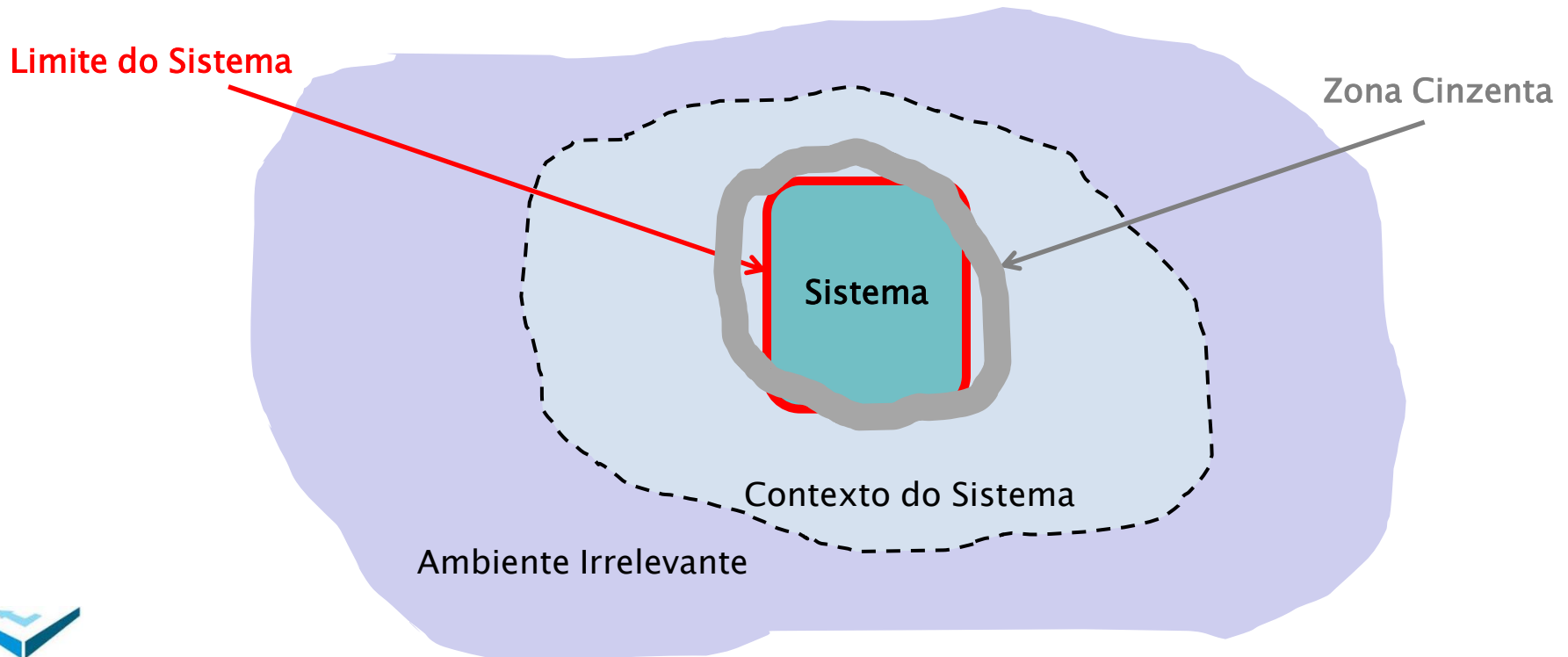
**Limite do Contexto**





## Zona Cinzenta entre o Sistema e o Contexto do Sistema

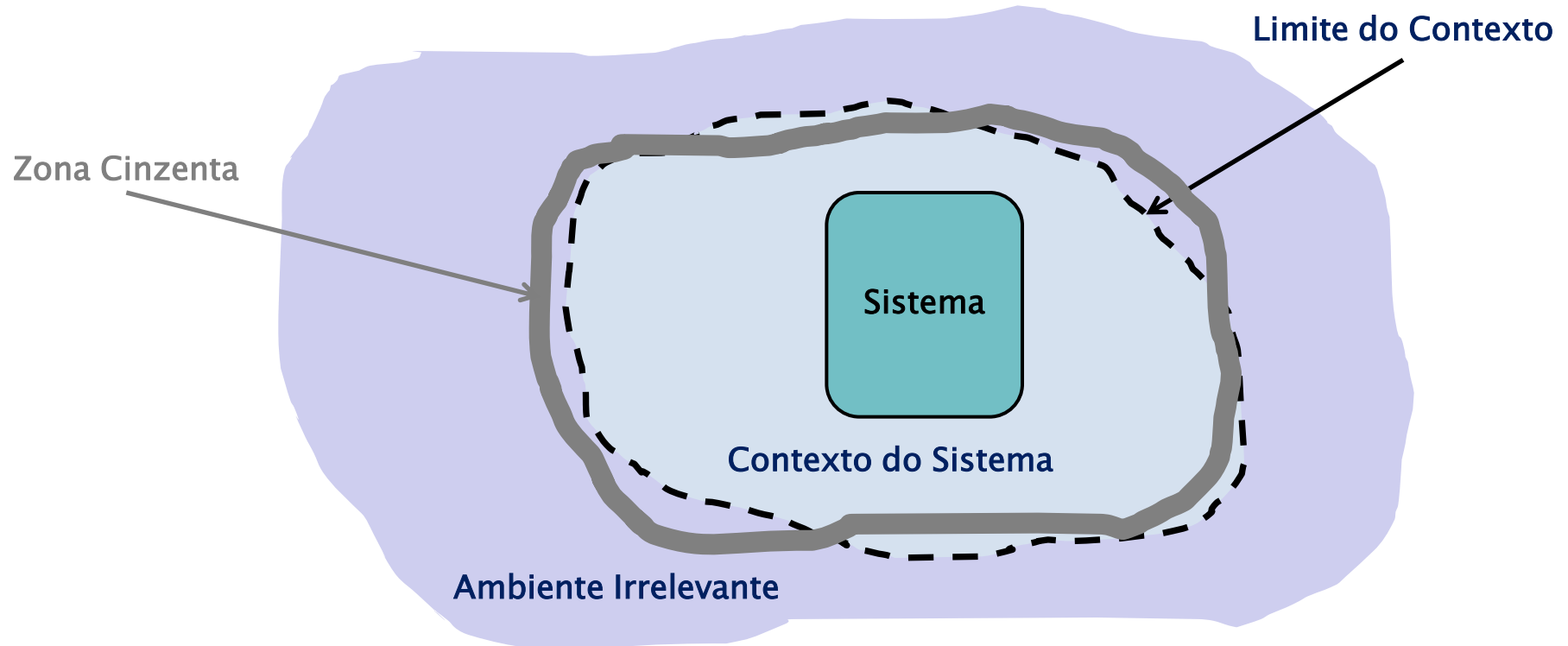
- Corresponde à separação inicialmente vaga (indefinida) do sistema e seu contexto;
- O limite do Sistema geralmente não é bem definido até o final do processo de Engenharia de Requisitos;
- Um bom processo de Engenharia de Requisitos deve resolver (ou eliminar) a zona cinzenta (eliminar dúvidas e fechar o escopo do software).





## Zona Cinzenta entre o Contexto e o ambiente irrelevante do Sistema

- Uma delimitação completa e precisa do Limite do Contexto é praticamente **impossível**;
- **Nem sempre é possível** visualizar aspectos do ambiente que influenciam ou não o sistema.

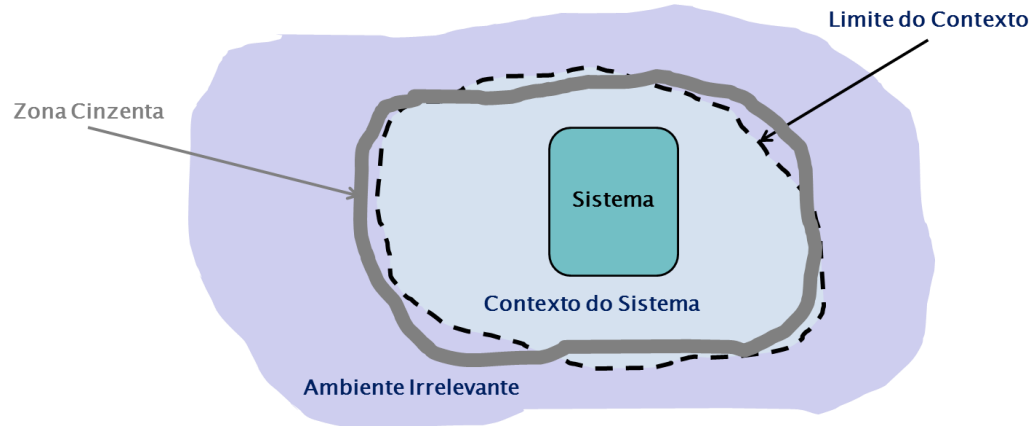






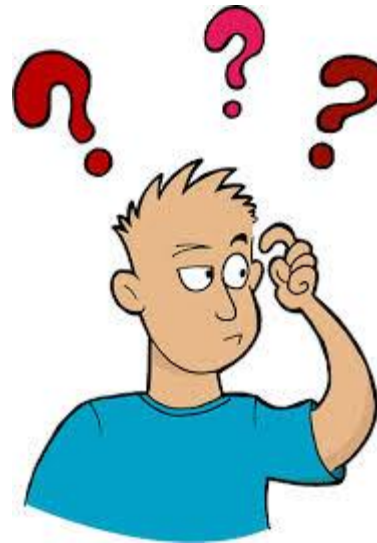
## Zona Cinzenta entre o Contexto e o ambiente irrelevante do Sistema

- Diferentemente da zona cinzenta entre o Sistema e o Contexto, a zona cinzenta entre o Contexto e o Ambiente Irrelevante **não** precisa ser necessariamente resolvida.





# Como documentar o Contexto de um sistema ?

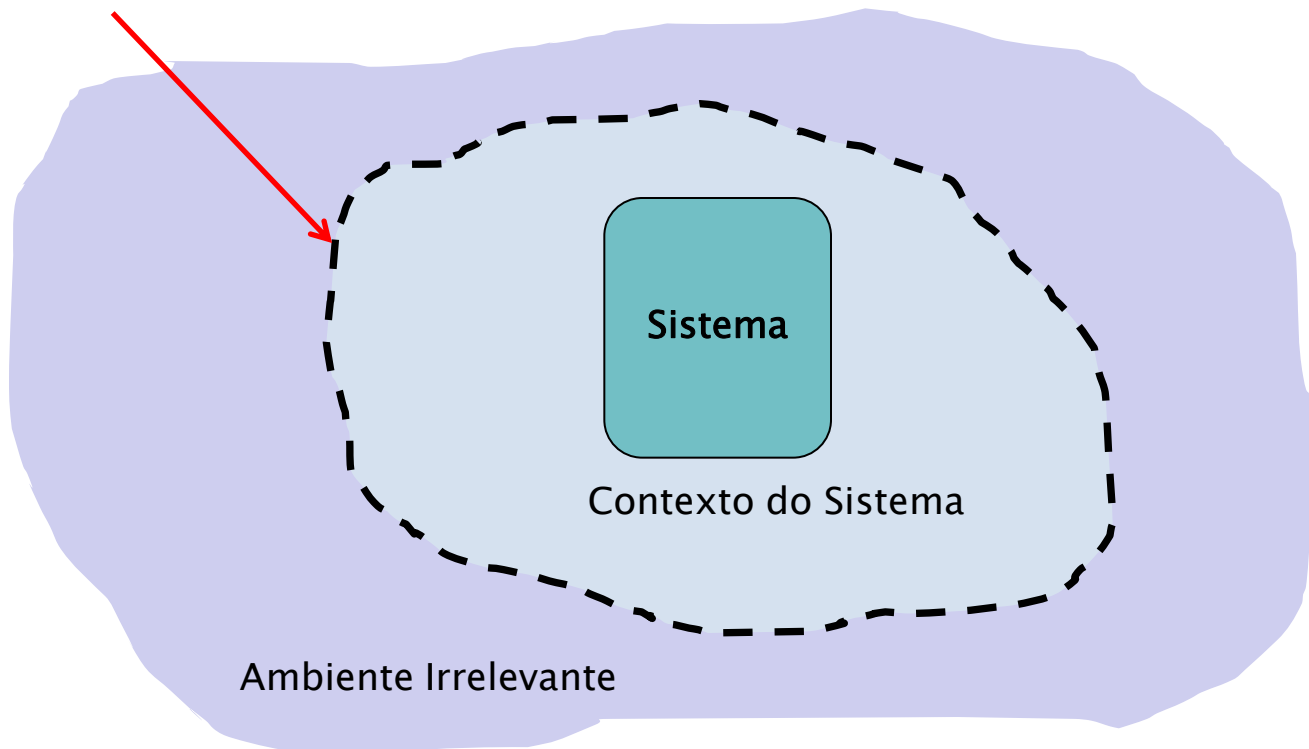




# Documentação

- Diagramas de Casos de Uso (UML)
- Diagramas de Contexto (Análise Estruturada)

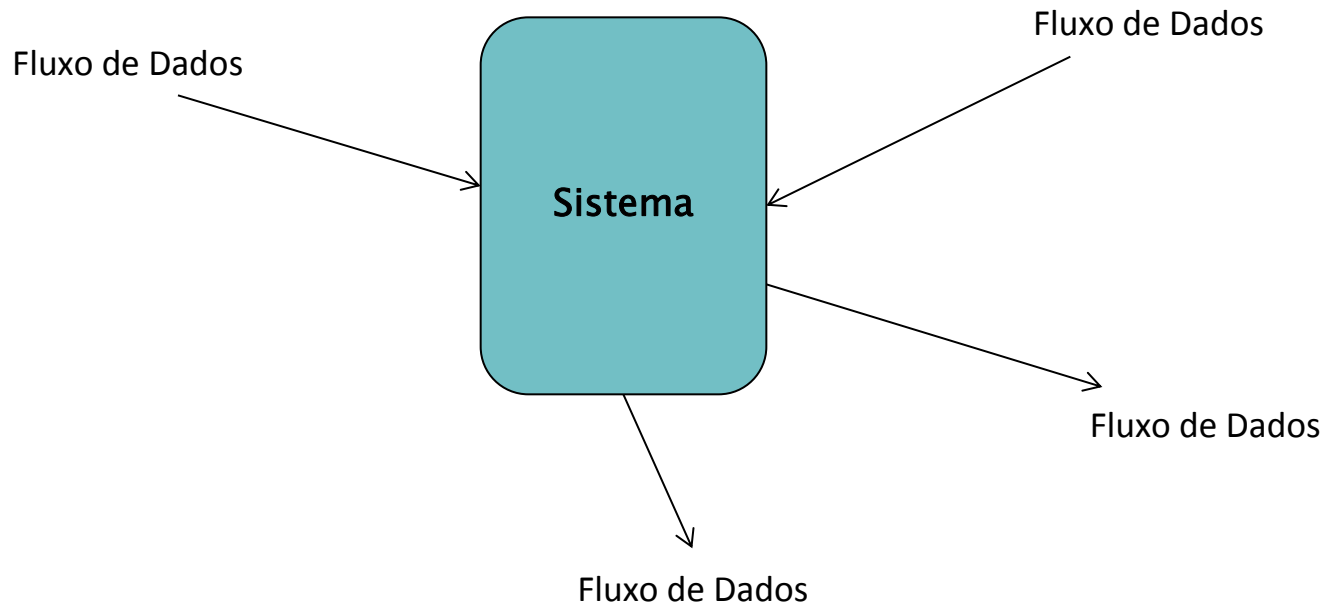
Limite do Contexto





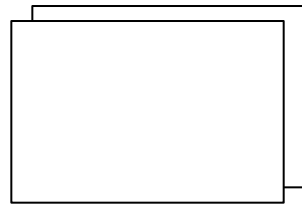
# Diagrama de Contexto

- O **Diagrama de Contexto** é um caso especial do Diagrama de Fluxo de Dados (Análise Estruturada) no qual uma única bolha representa o sistema inteiro;
- Define a interconexão do sistema com o ambiente no qual está inserido;
- As pessoas, organizações ou sistemas com as quais o sistema a ser desenvolvido se comunica, são denominadas Entidades Externas.





## Elementos do Diagrama de Contexto



**Entidade Externa:** Fonte de informações para o sistema ou destino de informações geradas pelo sistema



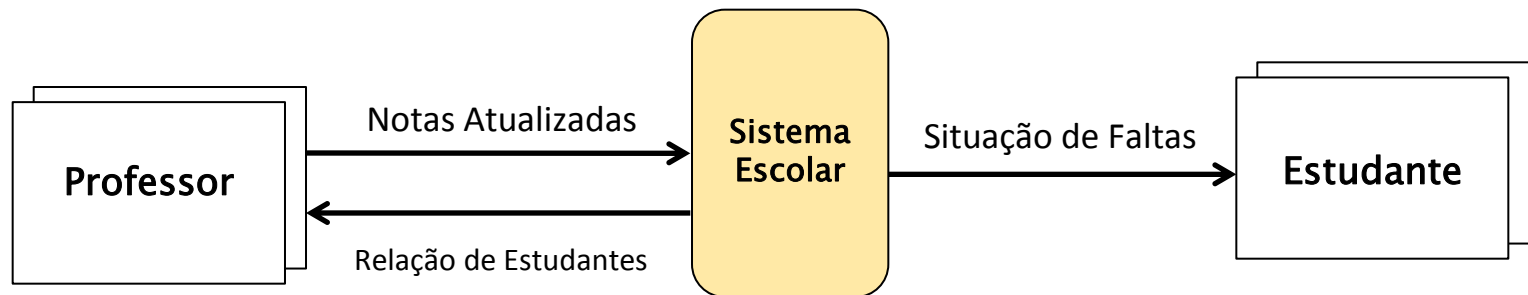
**Fluxo de Dados**



**Sistema  
(processo)**

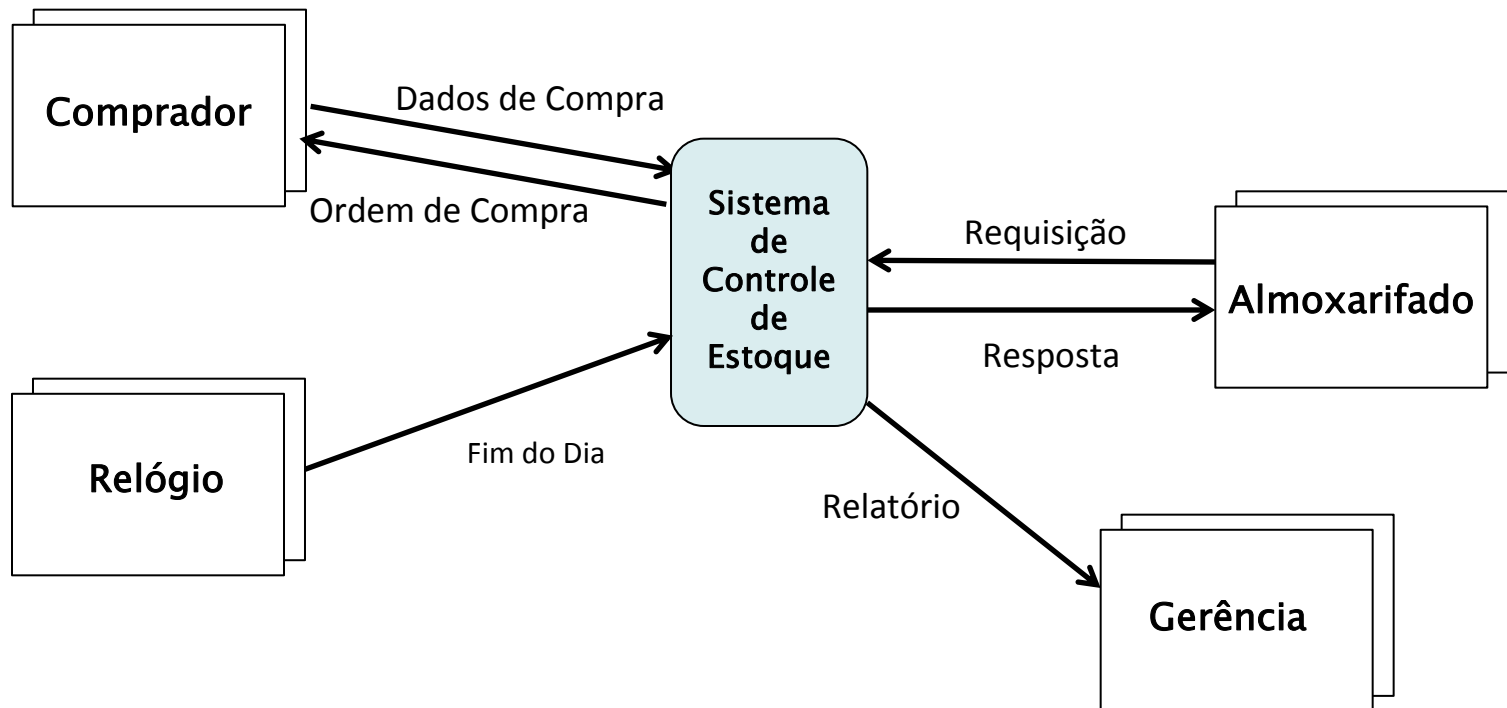


## Diagrama de Contexto





## Diagrama de Contexto





O que significa Elicitar Requisitos ?







# Elicitação de Requisitos

- Tarefa de identificar, descobrir, tornar explícito, obter o máximo de informações para o conhecimento do software a ser desenvolvido;
- Trás à tona os desejos e necessidades das partes interessadas;
- Auxilia a identificar o contexto do sistema.





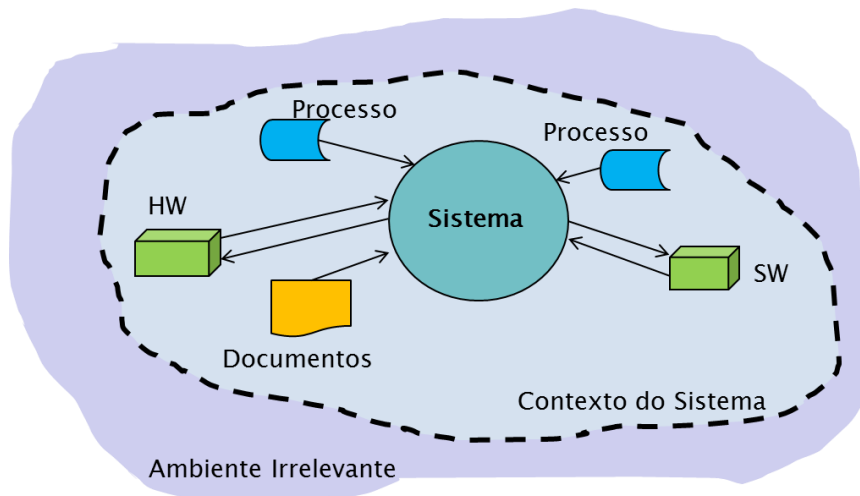
A elicitação de requisitos é feita a partir de que conhecimento ?





# Elicitação de Requisitos

- A base para a elicitação de requisitos é formada pelo conhecimento do **CONTEXTO** do software a ser desenvolvido, obtido durante a Engenharia de Requisitos.





# Fontes de Requisitos – Stakeholders



Stakeholders

- Partes Interessadas (Stakeholders)
- Documentos
- Sistemas em operação





# Como identificar os stakeholders ?





## Levantamento de Stakeholders

- Atividade que consiste em identificar os stakeholders relevantes (gerentes, especialistas do domínio, por exemplo);
- Acatar recomendações desses para identificar outros stakeholders.





# Como lidar com os stakeholders ?





# Interação com os Stakeholders

- Projetos complexos e “difíceis” podem envolver muitos stakeholders;
- Documentar stakeholders com informações tais como: nome, função (papel), dados pessoais, relevância do stakeholder, área, nível de expertise, objetivos e interesses no projeto, etc.
- Stakeholders que não recebem a devida atenção do Engenheiro de Requisitos **podem assumir postura crítica** em relação ao projeto.
- Stakeholders **podem demonstrar desmotivação** por estarem confortáveis com o sistema legado.
- Recomenda-se **firmar acordos formais**, para se evitar mal-entendidos e disputas sobre competências.







# Direitos e Deveres do Engenheiro de Requisitos



- Falar a linguagem dos Stakeholders;
- Conhecer o domínio da aplicação;
- Ser capaz de documentar os resultados do trabalho (gráficos e diagramas);
- Manter relacionamento respeitoso com os stakeholders;
- Apresentar ideias, alternativas, bem como resultados;
- Aceitar sugestões dos stakeholders para simplificar e facilitar o uso da aplicação;
- Assegurar que o sistema atenda às exigências funcionais e de qualidade dos stakeholders.



# Direitos e Deveres dos Stakeholders



- Auxiliar a inclusão do Engenheiro de Requisitos no domínio da aplicação;
- Suprir o Engenheiro de Requisitos com requisitos;
- Tomar decisões em tempo hábil;
- Respeitar estimativas de custo e viabilidade feitas pelo ER;
- Priorizar requisitos;
- Inspecionar os requisitos que o ER documenta (como protótipos, etc);
- Comunicar imediatamente mudanças de Requisitos;
- Respeitar o processo de Engenharia de Requisitos implementado.