



Unidade 4 – Princípios de Engenharia de Requisitos

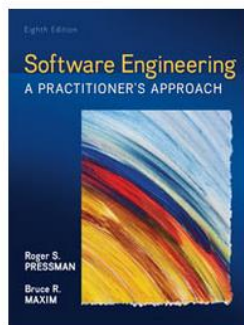


Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP

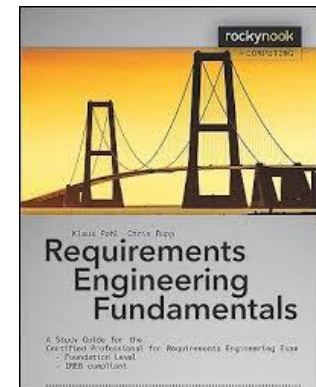
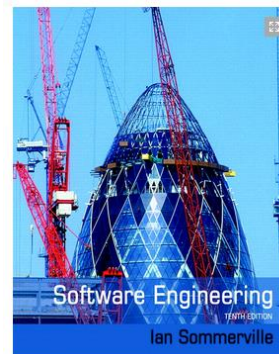


Bibliografia

- **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 – 10th 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





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





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





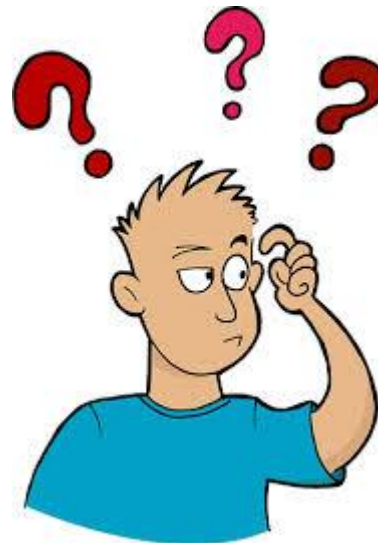
Em que situações um projeto de software é bem-sucedido ?

- ✓ 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 ?





Será que todos os projetos de Software são assim ?



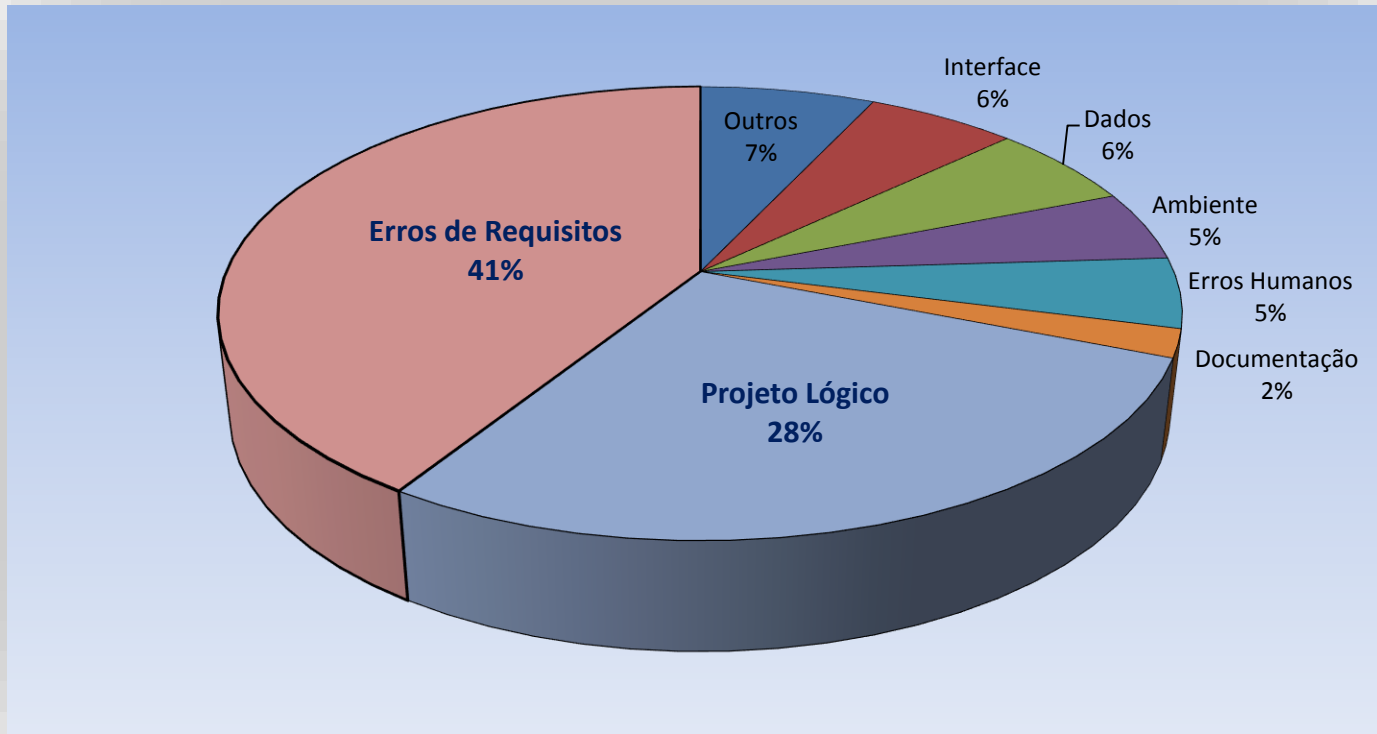
- Estudos mostram que 40% a 60% dos problemas ocorridos em projetos de software, são decorrentes de falhas no processo de requisitos.



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



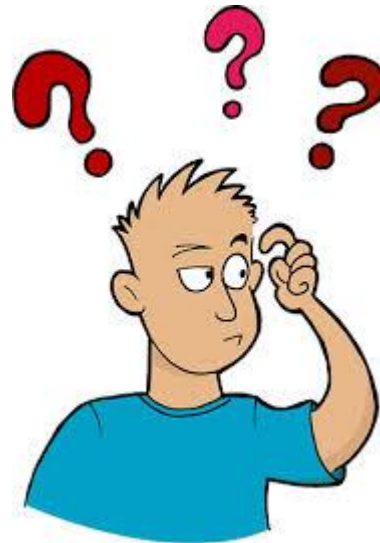
Erros de Projetos



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 usuário.
- 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 usuário;
- Levantar adequadamente as necessidades do usuário;
- Definir requisitos em conjunto com usuário (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

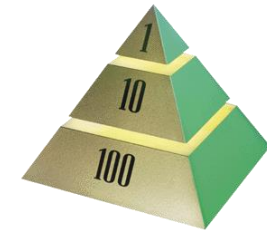
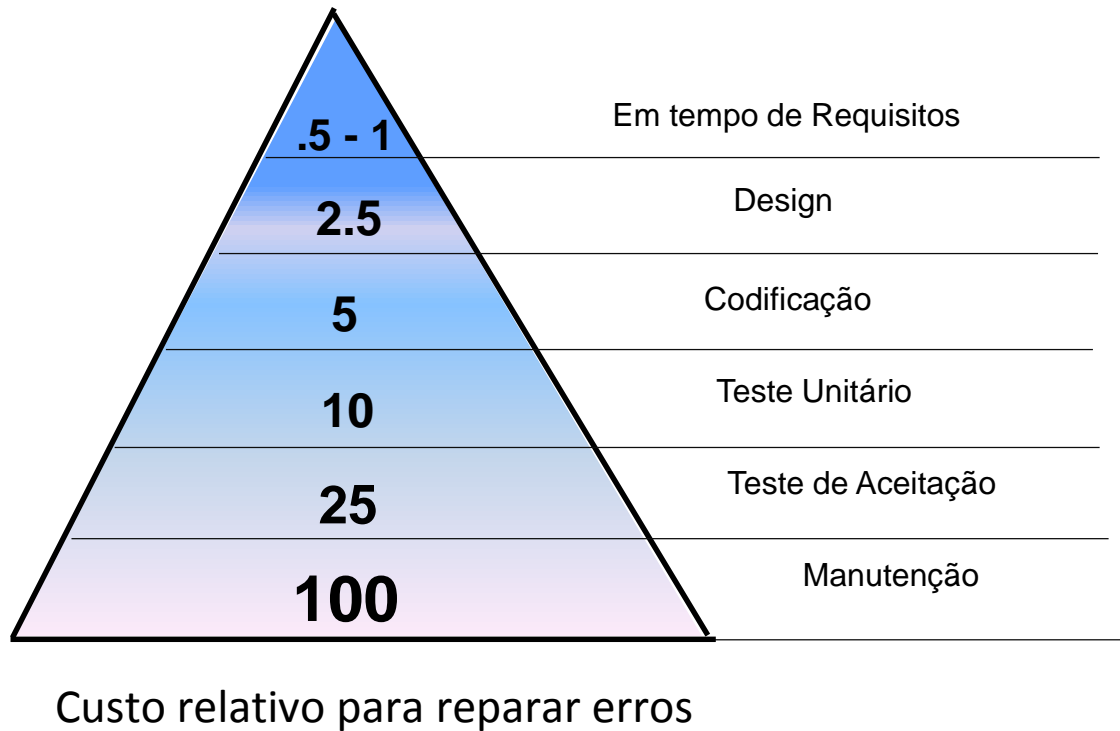
Propagação dos custos





O alto custo dos Requisitos Errados

A regra do 1-10-100

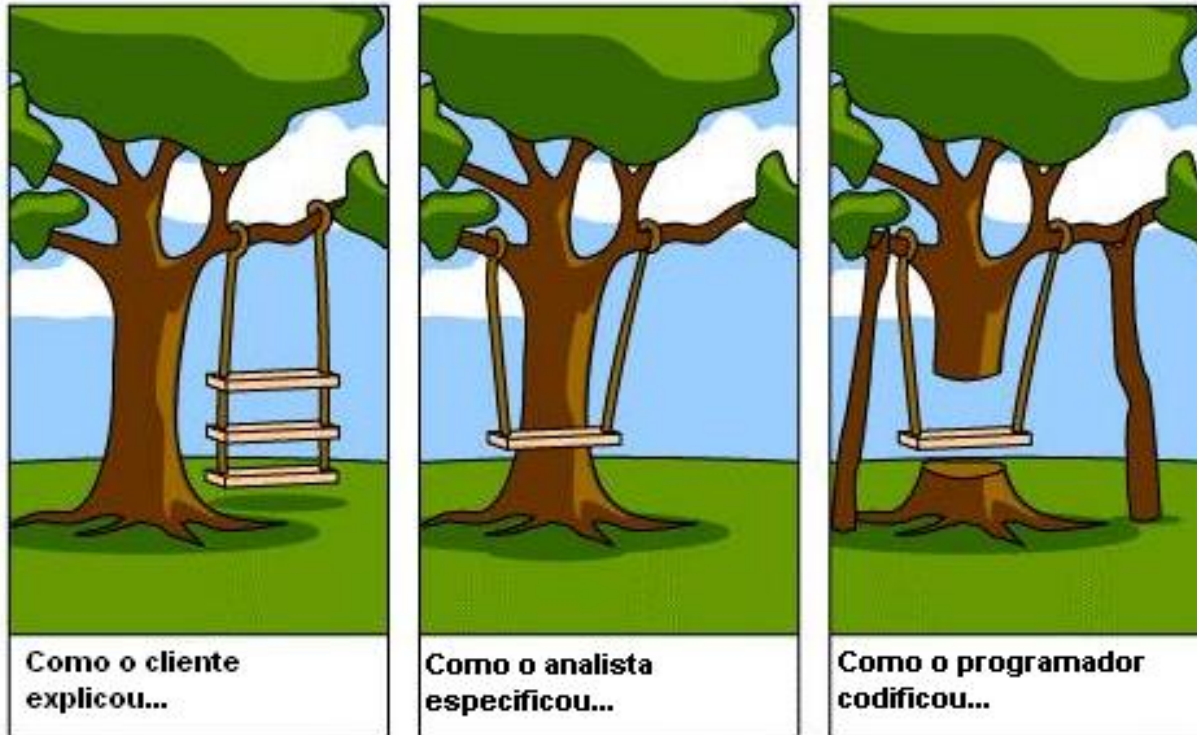


“Os resultados mostram como corrigir erros encontrados nos requisitos custa até 200 vezes menos do que em estágios mais avançados do ciclo de vida”

Fonte: Boehm 1988



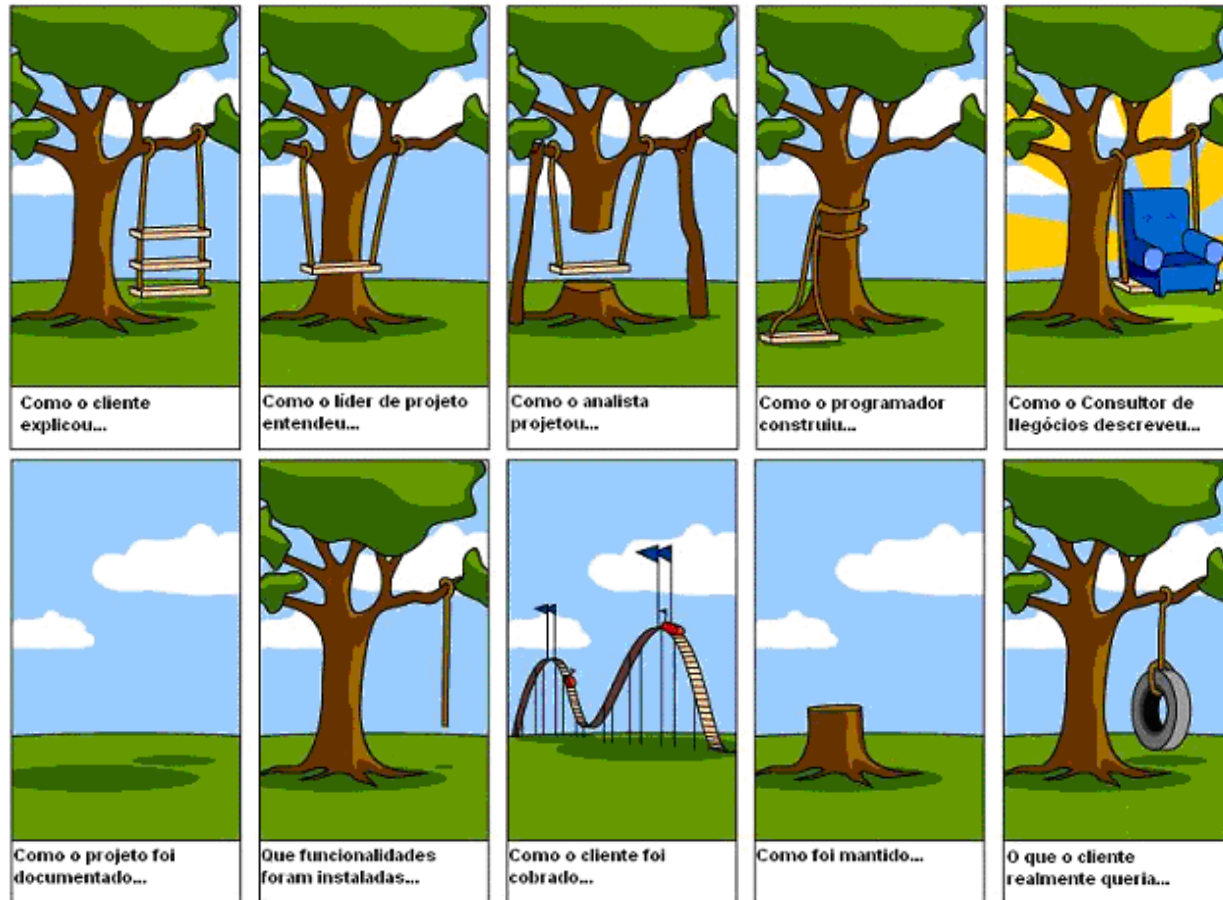
Descoberta de Requisitos



Problemas de Definição de Requisitos existem desde o início do Desenvolvimento de Software.



Descoberta de Requisitos



Esses problemas persistem até os dias de hoje !



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 usuário 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



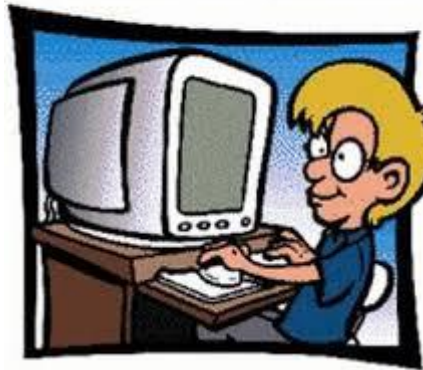


Problema da Pedra

O usuário diz:

Por favor, traga-me uma pedra !

Usuário





Problema da Pedra

Você lhe entrega a pedra !





Problema da Pedra

Usuário agradece, mas – desapontado - lhe diz !

Grato, mas na verdade o que eu queria era uma
pequena pedra vermelha !!!





Problema da Pedra

Você lhe diz: Puxa, mas porque você não me disse ?





Problema da Pedra

Usuário responde: Ué, mas você não perguntou ?





Problema da Pedra

Você lhe diz: Ok, aqui está sua pedra vermelha !





Problema da Pedra

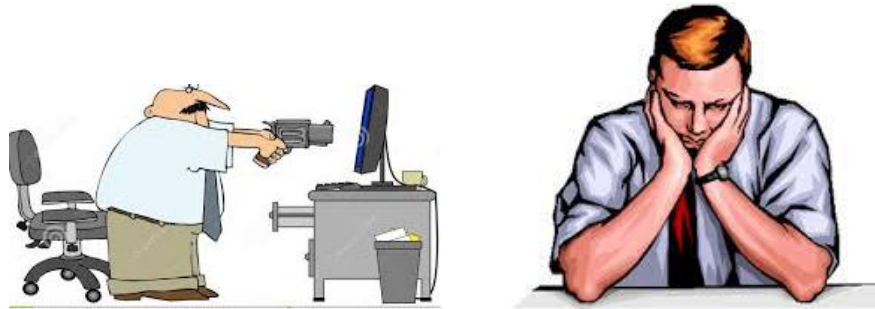
Usuário agradece, mas – ainda mais desapontado - lhe diz !

Grato, mas na verdade o que eu queria era uma pequena pedra vermelha, com formato elíptico !!!





Final do Filme



Usuário e Desenvolvedor frustrados !!!

Prazos comprometidos !!!

Custos elevados !!!



Problemas adicionais

- ✱ Em projetos de software, em geral, há times de software, o que representa envolvimento de mais pessoas.
- ✱ Em projetos reais de software, os prazos impossibilitam a estratégia de “tentativa e erro”.





Um simples exemplo





Desenvolvimento de uma
aplicação que calcula o
perímetro de um triângulo !





Programa para cálculo do Perímetro de um Triângulo

- O usuário deve informar o valor (em metros) de cada um dos lados, com precisão de duas casas decimais.
- A fórmula a ser utilizada é: **Lado1 + Lado2 + Lado3**
- O valor calculado deve ser informado também com precisão de duas casas decimais.
- O cálculo deve ser apresentado após comando do usuário.



Essa especificação é suficiente ?





Programa para cálculo do Perímetro de um Triângulo

Observações



- O que fazer quando o usuário entrar com valores nulos que representam **Lado1**, ou **Lado2** ou **Lado3** ?
- O que fazer quando o usuário entrar com valores negativos que representam **Lado1**, ou **Lado2** ou **Lado3** ?
- O que fazer quando o usuário não entrar com algum valor que representa **Lado1**, ou **Lado2** ou **Lado3** ?
- Quais os valores máximos que o usuário pode entrar para **Lado1**, ou **Lado2** ou **Lado3** ?





Programa para cálculo do Perímetro de um Triângulo

Observações

- O que fazer quando o usuário entrar com um valor errado para um dos lados **Lado1**, ou **Lado2** ou **Lado3** ? Existe no programa alguma opção para limpar os dados e reentrar ?
- O programa deve ser executado em que plataforma operacional ?
- O programa deve ser executado em interface texto ?
- O usuário pode executar o programa sem o mouse ?
- O programa deve ser executado em uma interface gráfica ?





Programa para cálculo do Perímetro de um Triângulo

Observações

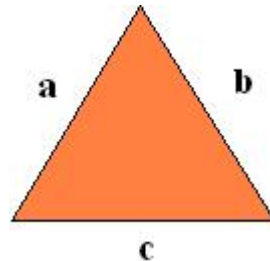
- O programa deve salvar o resultado em alguma área (persistência)?
- Que mensagens o programa deve enviar ao usuário quando houver algum tipo de erro ?
- O programa deve imprimir o resultado do cálculo ou apenas exibir numa janela de texto ou interface gráfica ?
- O que o programa deve fazer quando o usuário entrar com valores que não representam um triângulo ? (Exemplo: Lado1 = 1, Lado2 = 2 e Lado3 = 500)





Condição para a existência de um triângulo

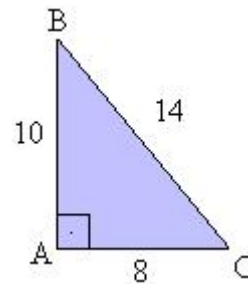
- Para construir um triângulo não se pode utilizar qualquer medida.
- Deve haver uma condição de existência.
- Para se construir um triângulo é necessário que a medida de qualquer um dos lados seja menor que a soma das medidas dos outros dois e maior que o valor absoluto da diferença entre essas medidas.)



$$\begin{aligned} |b - c| < a < b + c \\ |a - c| < b < a + c \\ |a - b| < c < a + b \end{aligned}$$



Condição para a existência de um triângulo – Exemplo



$$\begin{aligned} |b - c| &< a < b + c \\ |a - c| &< b < a + c \\ |a - b| &< c < a + b \end{aligned}$$

$$14 - 8 < 10 < 14 + 10$$

$$14 - 10 < 8 < 14 + 10$$

$$10 - 8 < 14 < 10 + 8$$



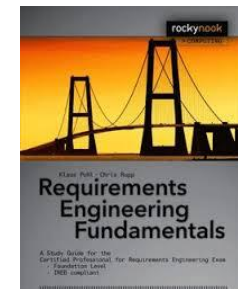
O que é Engenharia de Requisitos ?



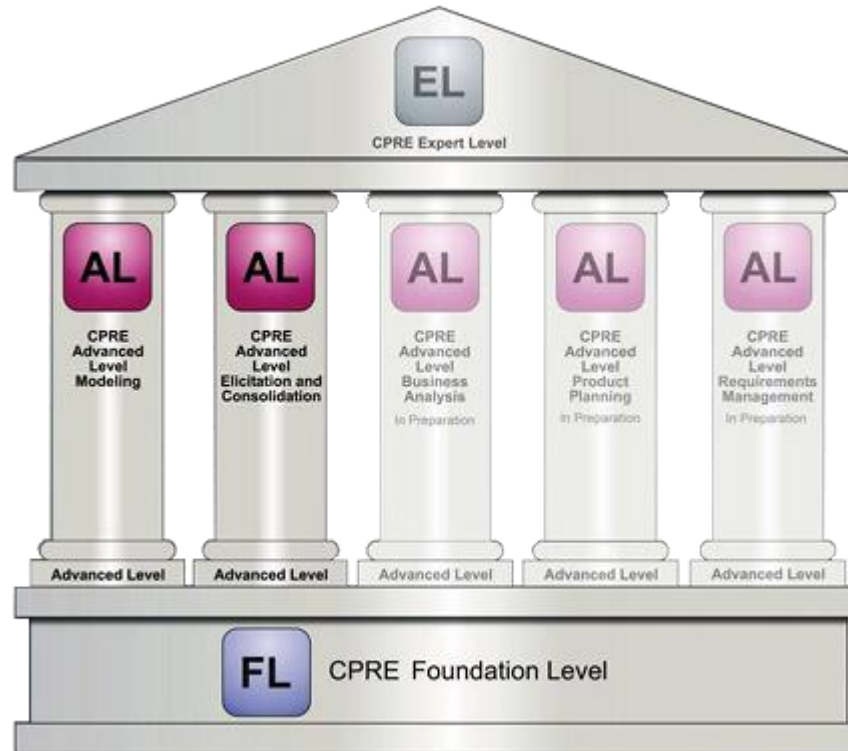


Engenharia de Requisitos

- ✓ 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.



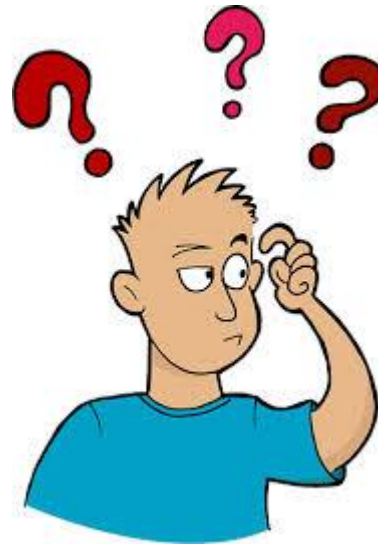
Fonte: Pohl, 1996



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



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





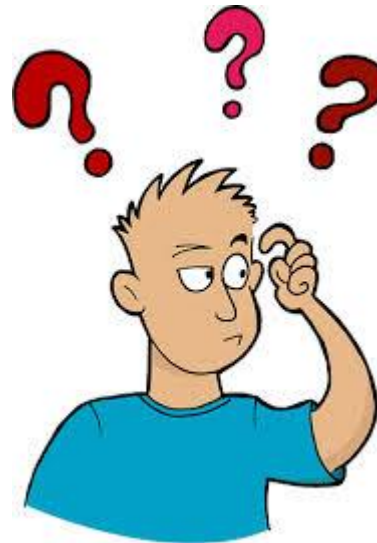
Definição – Stakeholders

- Um stakeholder de um sistema é uma pessoa ou uma organização que tem influência (direta ou indireta) nos requisitos de um sistema.





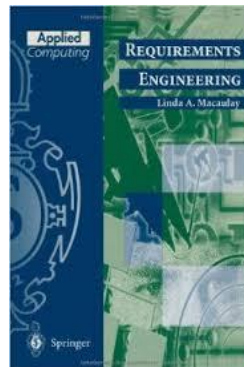
Qual o impacto no desenvolvimento do software quando não se leva em conta todos os Stakeholders ?





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 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)



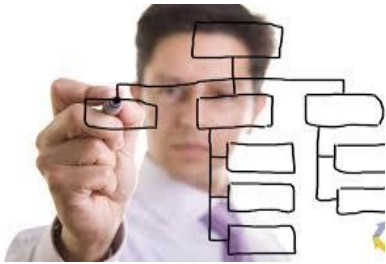
Qual o perfil de um Engenheiro de Requisitos ?





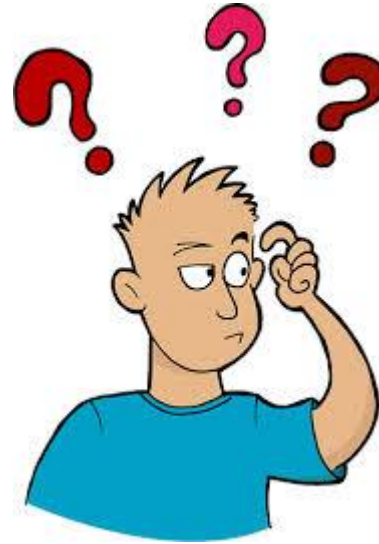
O Engenheiro de Requisitos

- **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 ?





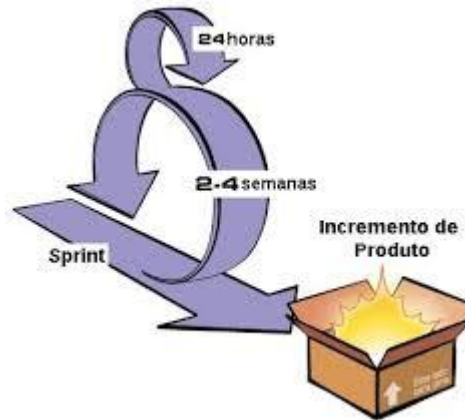
Engenharia de Requisitos no Modelo Cascata



- 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.



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 Usuários;
- 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

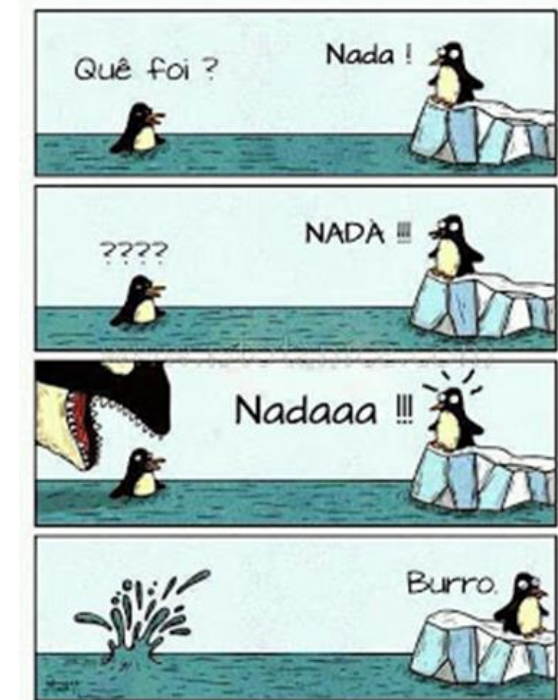
- Muitas vezes, as informações não são adequadamente transmitidas, ou nem mesmo transmitidas.
- O autor pode pressupor que o leitor tenha algum tipo de conhecimento implícito anterior sobre o tema.
- Essas simplificações podem causar problemas em termos de Requisitos, pois os Requisitos podem tornar-se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





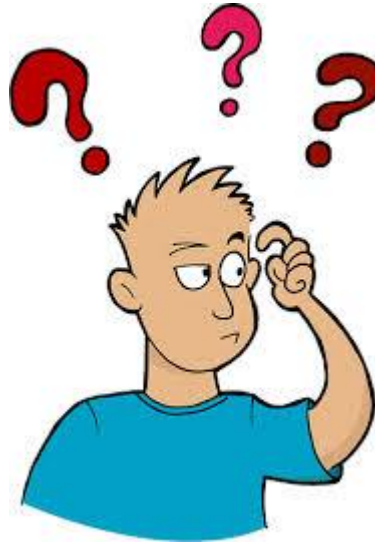
Ambiguidade

- 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 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 são cumpridas, pois meramente limitam o espaço da solução disponível durante o processo de desenvolvimento.





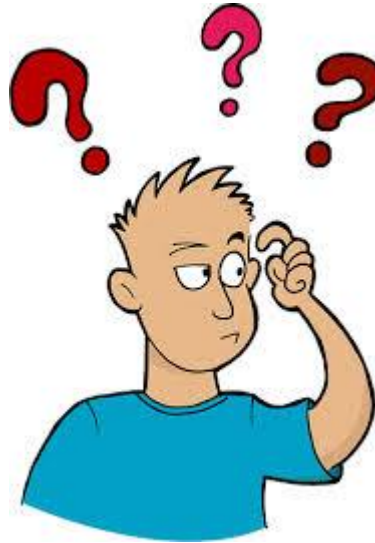
Os limites do Sistema

- 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

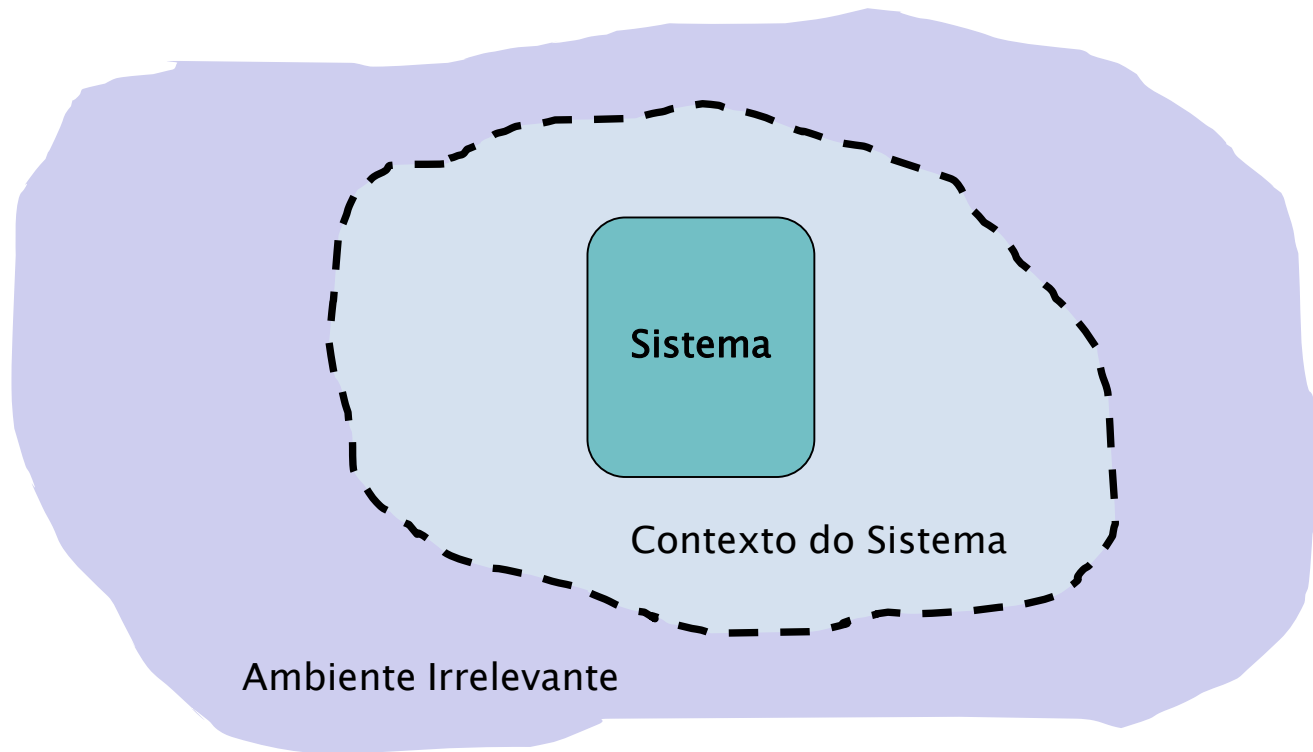
- É a parte do ambiente do sistema que é relevante para a definição e a compreensão dos requisitos de um sistema a ser desenvolvido.
- Os seguintes aspectos da realidade influenciam o contexto de um sistema:
 - ⊕ Pessoas (usuários ou grupos de usuários)
 - ⊕ Sistemas em operação
 - ⊕ Processos
 - ⊕ Eventos





Contexto do Sistema

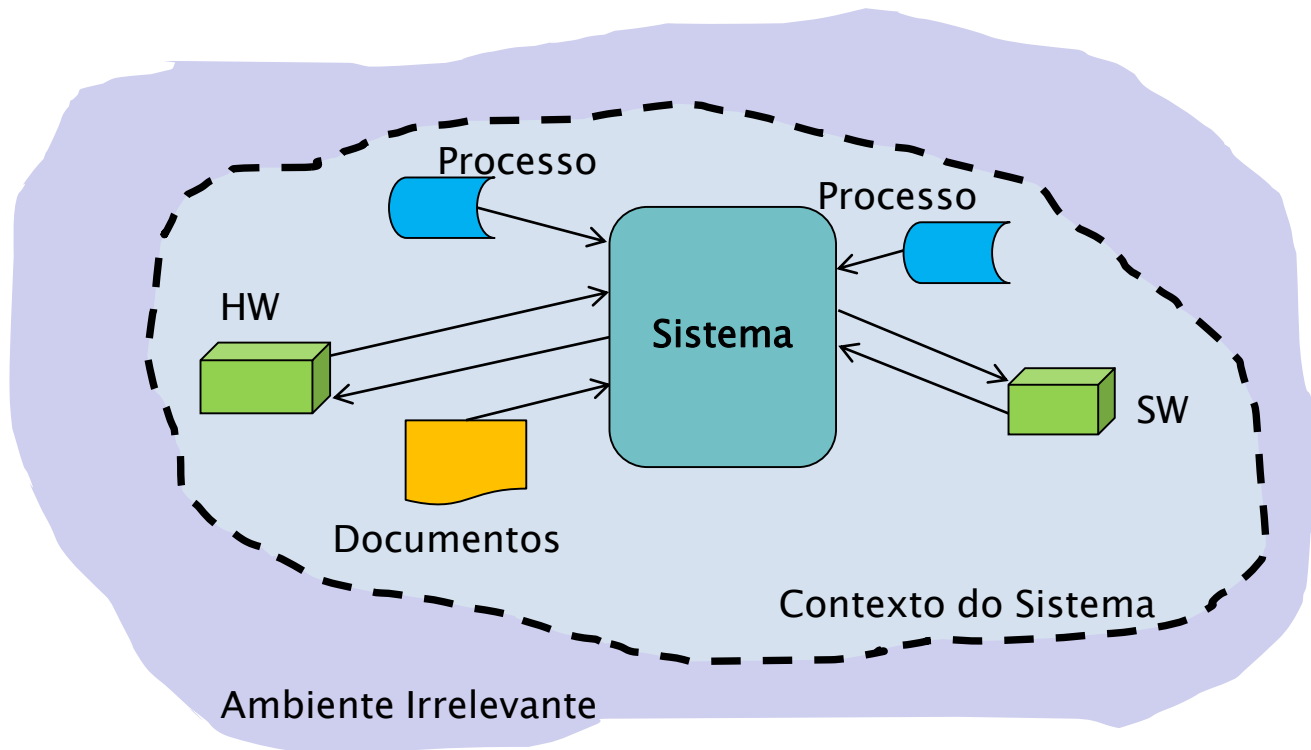
- É responsabilidade do Engenheiro de Requisitos definir adequadamente o contexto do sistema.





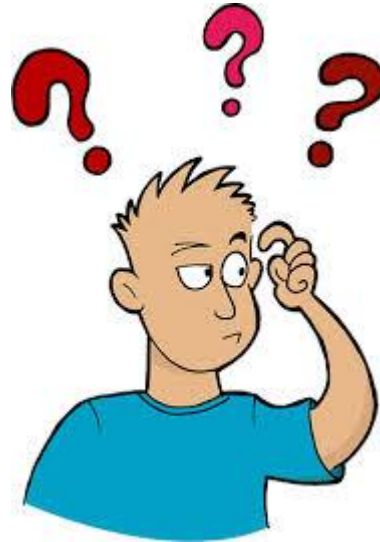
Contexto do Sistema

- 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.





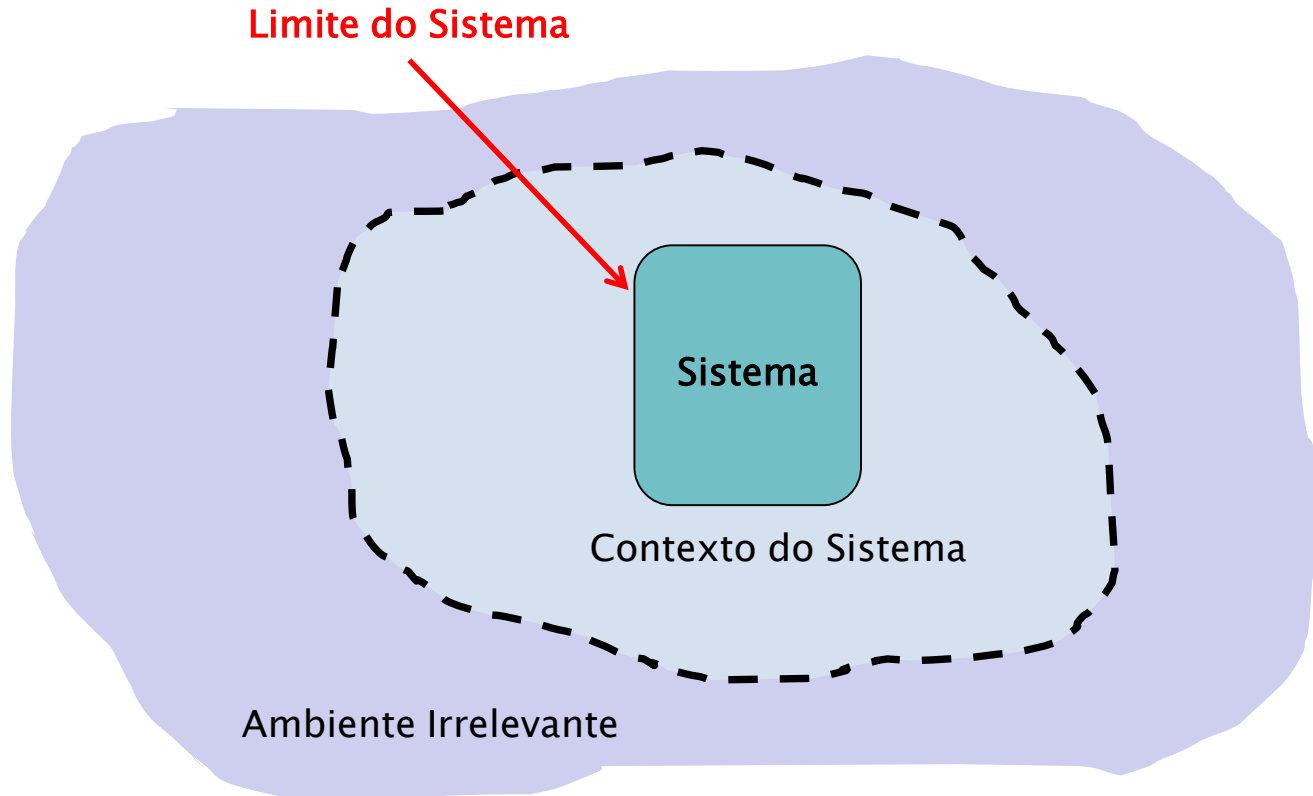
O que é limite do sistema ?





Limite do Sistema

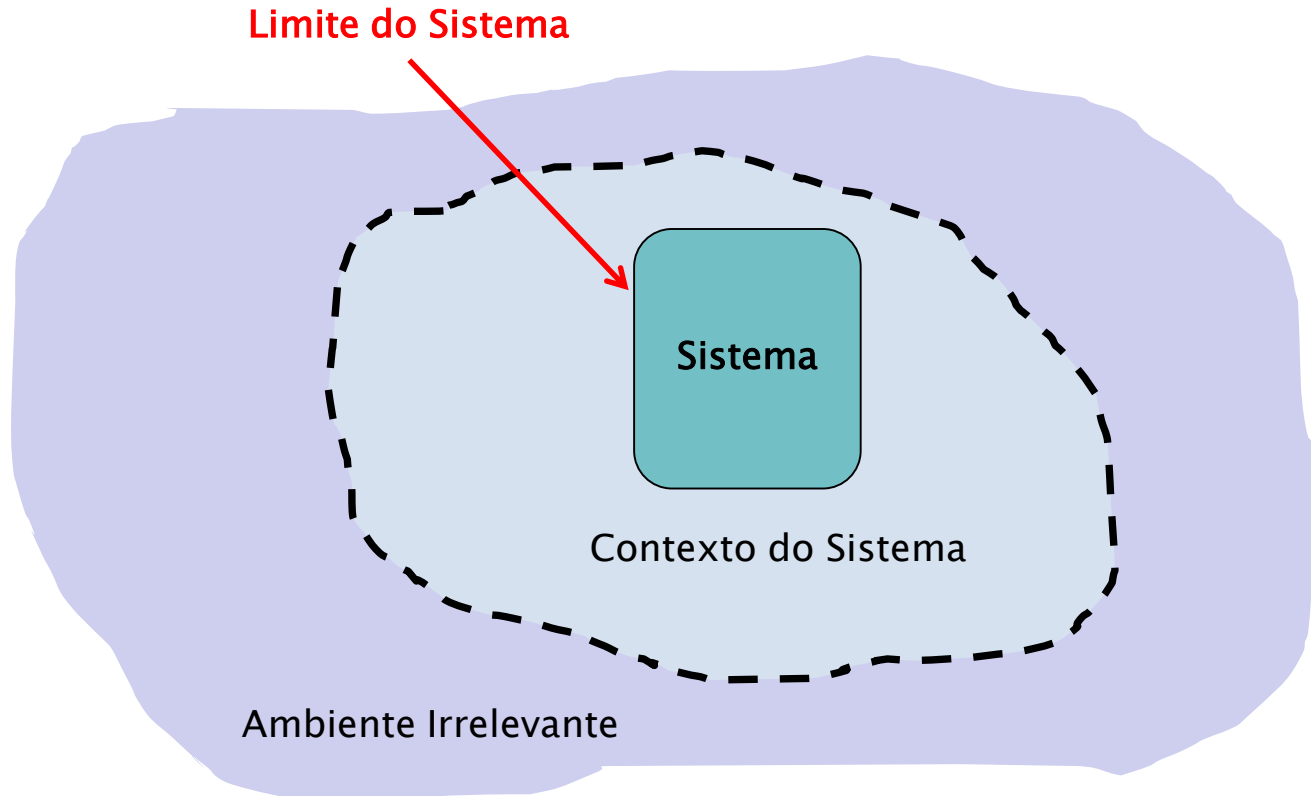
- O limite do sistema separa o sistema a ser desenvolvido do seu ambiente (isto é, separa a parte da realidade que pode ser modificada ou alterada pelo processo de desenvolvimento) daqueles aspectos do ambiente que não podem ser mudados ou modificados pelo processo de desenvolvimento.





Limite do Sistema

- Todos os aspectos que se encontram dentro do limite do sistema podem ser alterados durante o desenvolvimento do sistema.

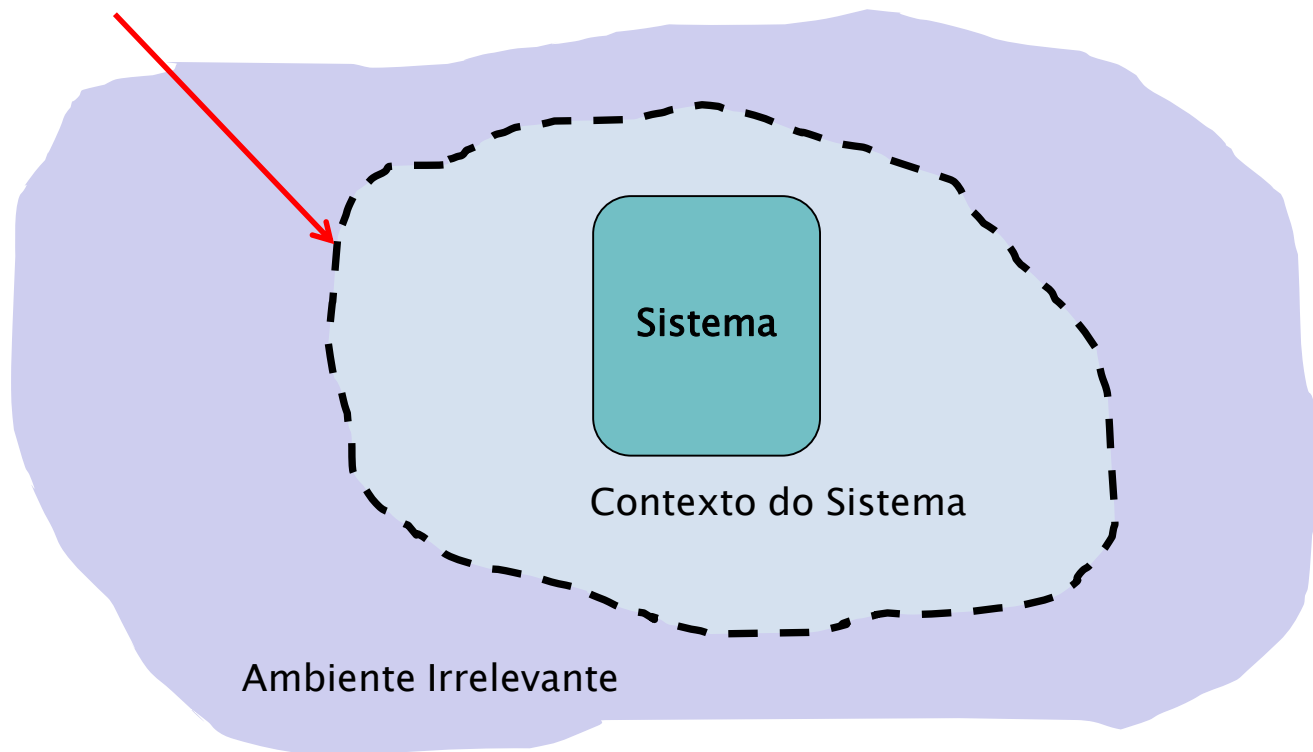




Limite do Contexto

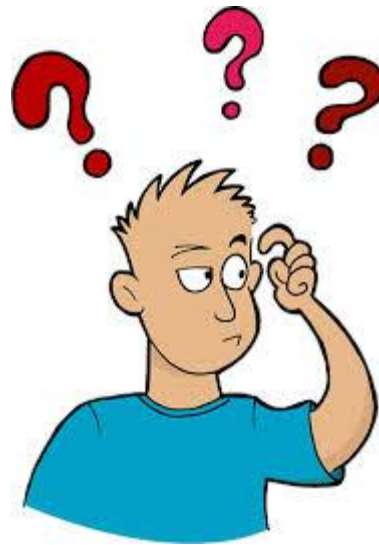
- O limite do contexto separa a parte relevante do ambiente de um sistema a ser desenvolvido da parte irrelevante, isto é, a parte que não influencia o sistema a ser desenvolvido e, sendo assim, não precisa ser considerado durante a Engenharia de Requisitos.

Limite do Contexto





Como documentar o Contexto de um sistema ?





Documentação

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

Limite do Contexto

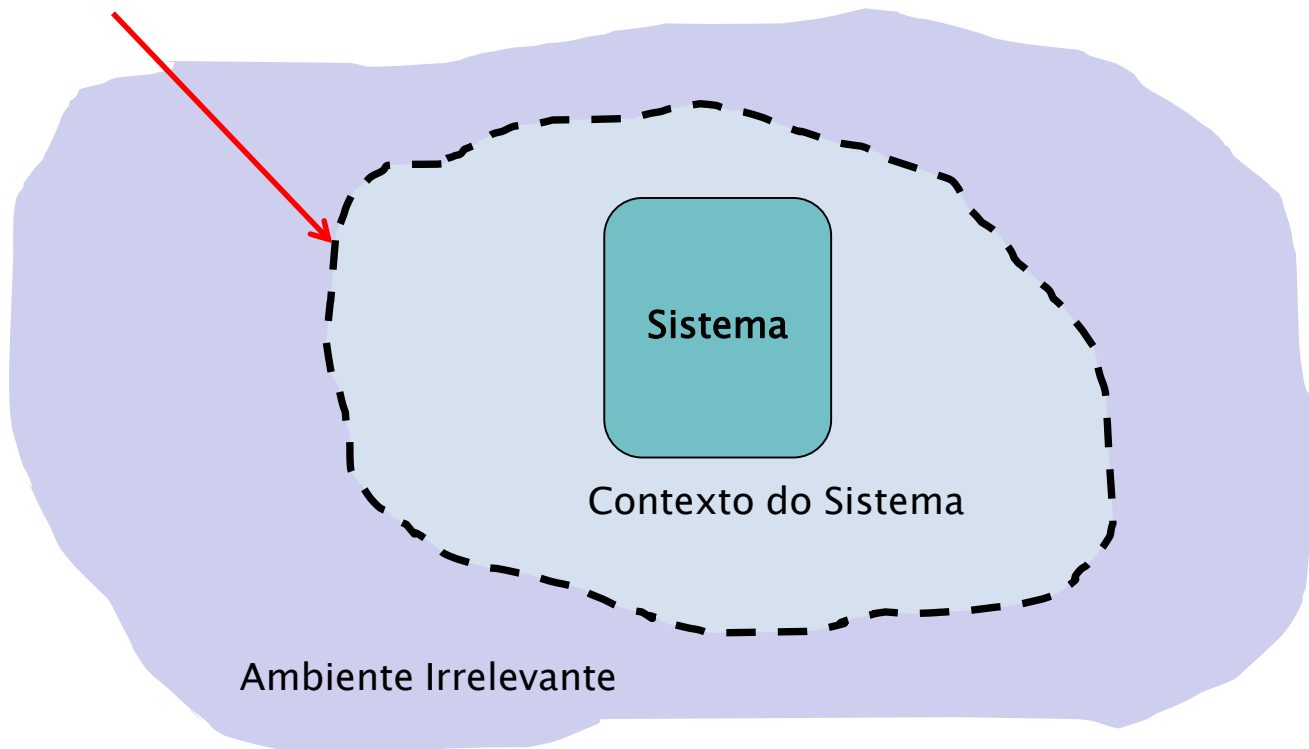
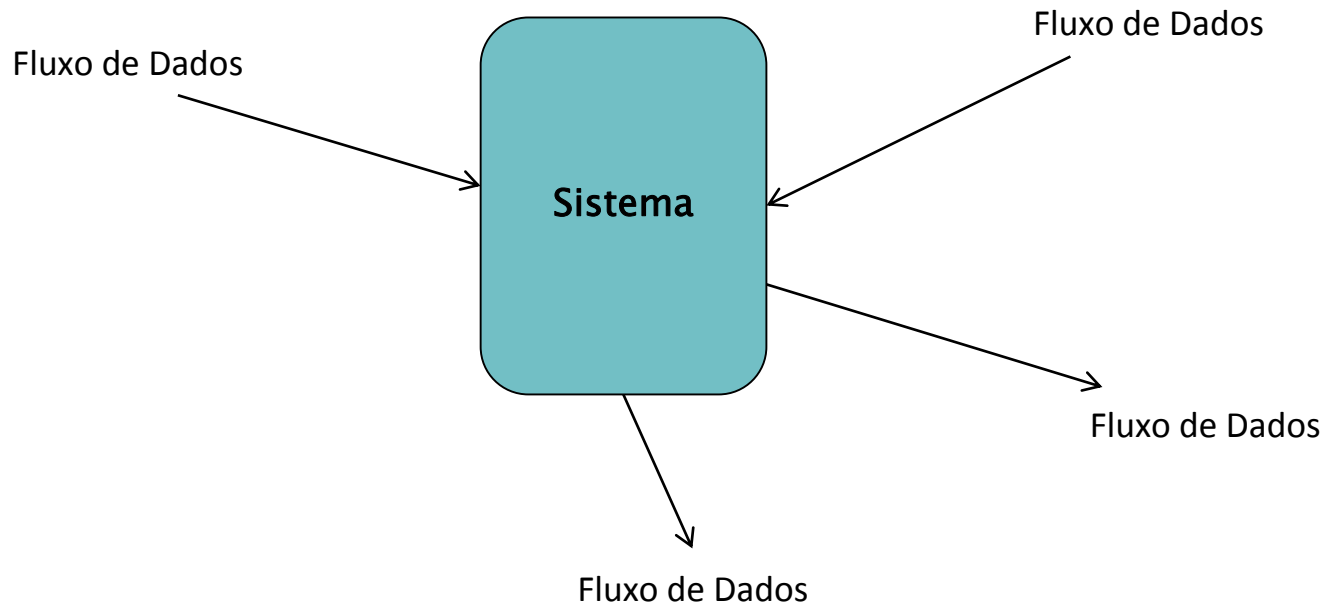




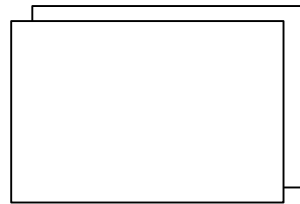
Diagrama de Contexto

- 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

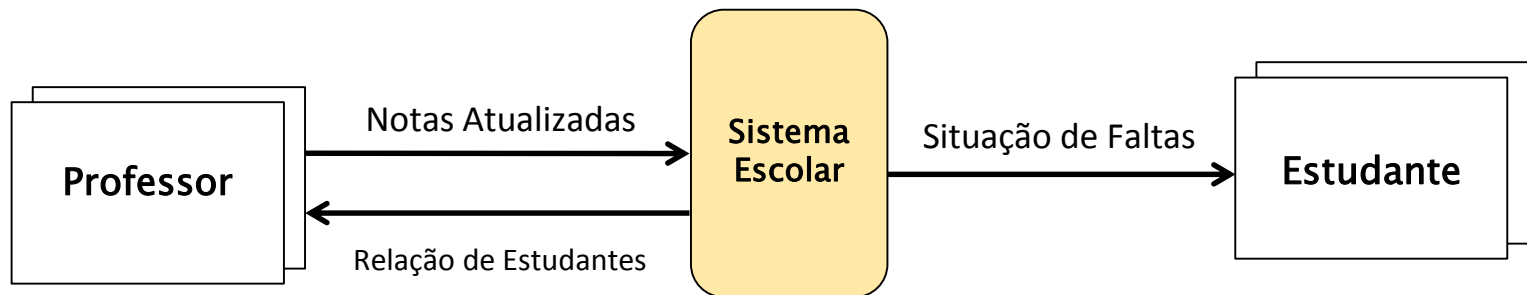
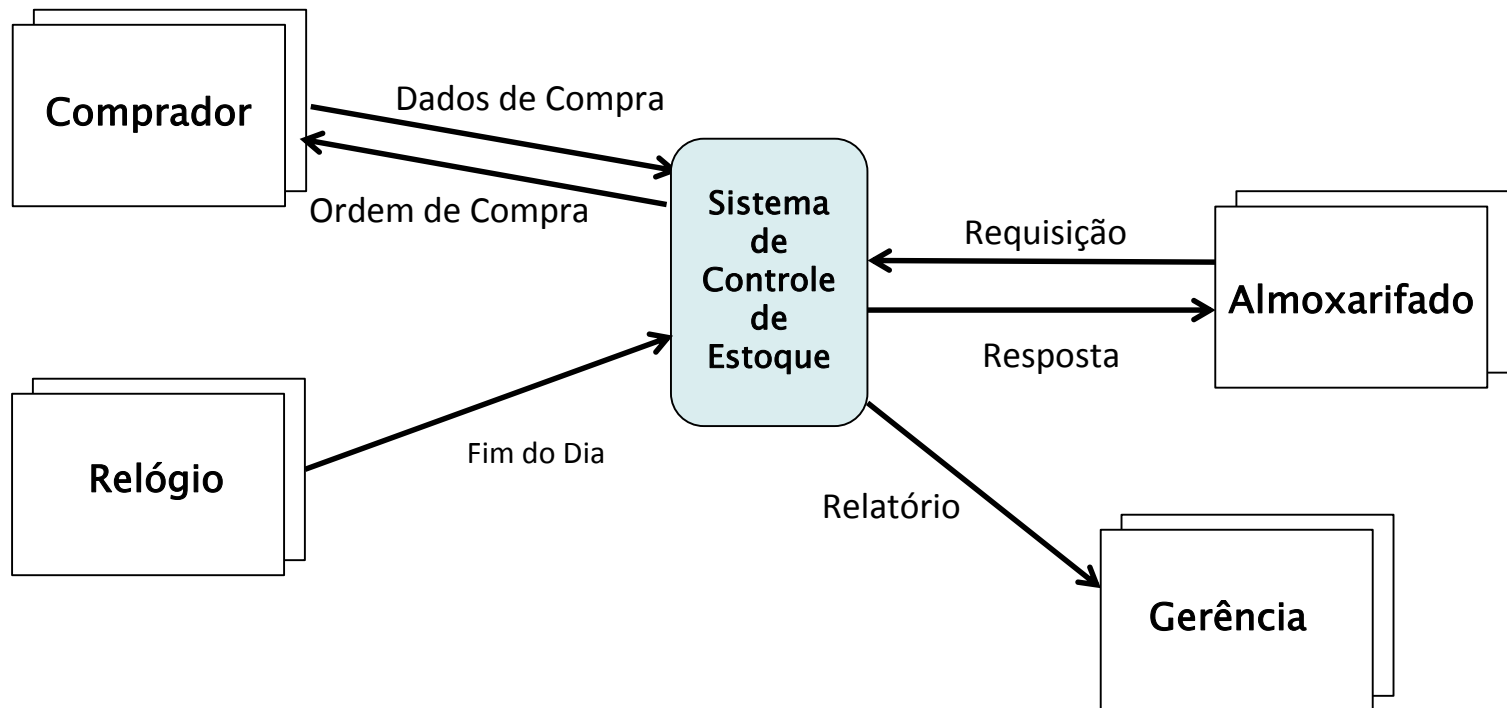




Diagrama de Contexto





O que significa Elicitar Requisitos ?





Elicitação de Requisitos

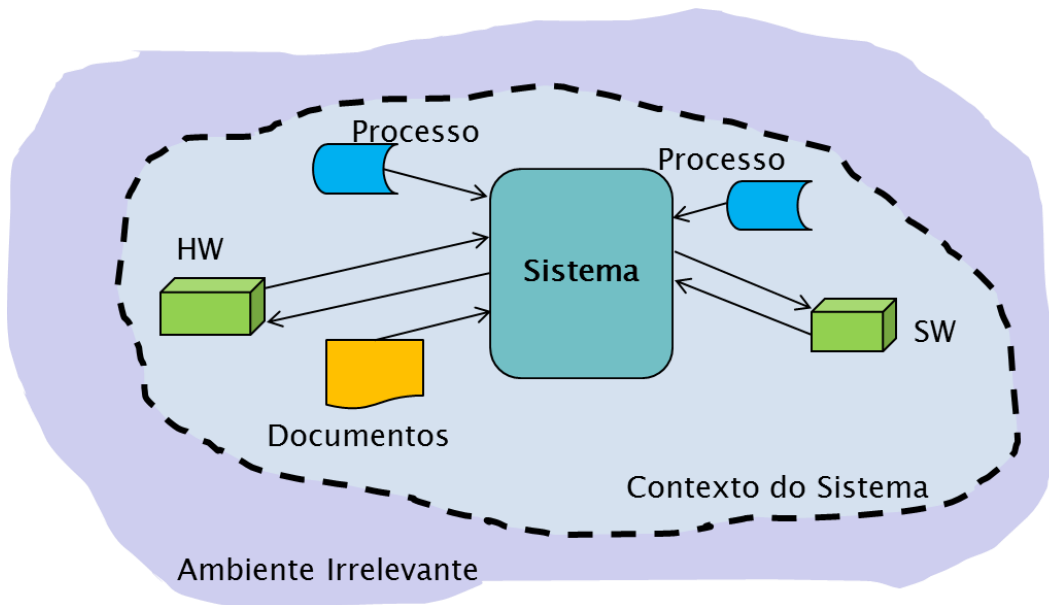
- Tarefa de identificar, descobrir, tornar explícito, obter o máximo de informações para o conhecimento do software a ser desenvolvido.





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



- Usuários
- Documentos
- Sistemas em operação





Como identificar os usuários ?





Usuários do Sistema

- Identificar os usuários relevantes (gerentes, especialistas do domínio, por exemplo)
- Acatar recomendações desses para identificar outros usuários.
- Com quem mais você acha que eu deveria falar ?



Fonte: Pressman



Como lidar com os usuários ?





Interação com os Usuários

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





Deveres do Engenheiro de Requisitos



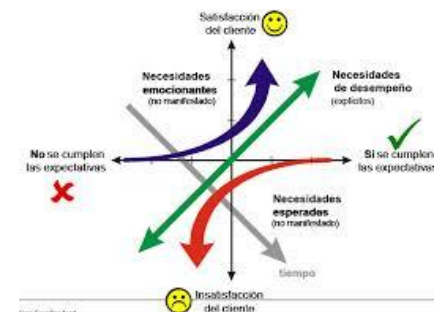
- Falar a linguagem dos Usuários;
- Conhecer o domínio da aplicação;
- Ser capaz de documentar os resultados do trabalho (gráficos e diagramas);
- Manter relacionamento respeitoso com os usuários;
- Avaliar sugestões dos usuários para simplificar e facilitar o uso da aplicação;
- Considerar diversos pontos de vista dos usuários (áreas de conflito);
- Assegurar que o sistema atenda às exigências funcionais e de qualidade dos usuários.

Fonte: Pressman



Técnicas de Elicitação

- Têm a finalidade de identificar os requisitos **subconscientes** (óbvios), **conscientes** (normais, esperados) e **inconscientes** (ocultos e inesperados) dos usuários. (**Modelo de Kano – Satisfação do Cliente**)



Fonte: Chris e Pohl



Técnicas de Pesquisa

- Baseiam-se em entrevistas ou questionários.
- Partem do pressuposto de que o respondente é capaz de expressar explicitamente seu conhecimento;
- Respondente está comprometido a investir tempo e esforço para a elicitación;
- São geralmente dirigidas pelo ER;
- ER que faz as perguntas;





Entrevistas

- ER faz as perguntas para um ou mais Usuários;
- ER documenta as respostas;
- Perguntas bem formuladas podem revelar requisitos;
- Principal desvantagem: tempo consumido para a elicitación.





Questionários

- ER pode fazer uso de questões abertas ou fechadas (ex. Múltipla escolha) ;
- Para grande número de Usuários, questionário online é uma opção viável;
- Podem elicitar grande número de informações em curto espaço de tempo;
- Baixo custo;
- Não fornecem retroalimentação imediata entre Usuário e ER.





Técnicas de Criatividade

- Tem a finalidade de desenvolver requisitos inovadores;
- Podem esboçar uma visão inicial do sistema;
- Podem elicitar fatores inesperados (fascinantes) de satisfação (delighters);
- Dentre as técnicas, cita-se: brainstorming, brainwriting (método 6-3-5, seis participantes escrevem 3 ideias em um papel, durante 5 rodadas).



Fonte: Chris e Pohl



Documentação de Requisitos

- Na ER, informações levantadas durante as diversas atividades devem, obrigatoriamente, ser documentadas.





Especificação de Requisitos

- É uma coleção de requisitos representada de forma sistemática, tipicamente para um sistema ou componente, atendendo a determinados critérios.
- Pode ser um documento escrito, um conjunto de modelos gráficos, um conjunto de cenários de uso, um protótipo ou qualquer combinação dos fatores citados.



Fonte: Pressman



Razões para se documentar Requisitos

- Requisitos formam a base para o desenvolvimento do sistema.
- Requisitos tem relevância legal
- Documentos de requisitos são complexos
- Requisitos devem ser acessíveis para todas as partes envolvidas





Tipos de Documentação

- Requisitos para um sistema podem ser documentados a partir de três perspectivas diferentes. (Estrutural, Funcional, Comportamental)
- Na prática, tanto linguagem natural quanto a modelagem conceitual são utilizadas para esse fim, ou muitas vezes, emprega-se uma combinação apropriada entre as duas.





Documentação de Requisitos com Linguagem Natural

- É a forma de documentação de requisitos mais aplicada na prática.
- Vantagem: Usuários não precisam aprender nova notação.
- Entretanto, pode resultar em requisitos ambíguos.





Documentação de Requisitos com Modelos Conceituais

- Estão associadas à cada perspectiva.
- Pode-se retratar os requisitos de forma mais compacta.
- Apresentam menor grau de ambiguidade.
- No entanto, exige conhecimentos específicos de modelagem.





Diagramas – Modelos Conceituais

- Diagrama de Caso de Uso.
- Diagrama de Classes.
- Diagrama de Atividades.
- Diagrama de Estados





Conteúdos Padrão Customizados

As seguintes questões deveriam ser abordadas por qualquer estrutura de documento de Especificação de Requisitos:



- ⊕ Introdução
- ⊕ Finalidade
- ⊕ Cobertura do Sistema
- ⊕ Stakeholder
- ⊕ Definições, acrônimo e abreviações
- ⊕ Referências
- ⊕ Visão Geral
- ⊕ Ambiente do Sistema
- ⊕ Descrição da Arquitetura
- ⊕ Funcionalidades do Sistema
- ⊕ Usuários e público-alvo
- ⊕ Restrições
- ⊕ Pressupostos
- ⊕ Requisitos
- ⊕ Apêndices
- ⊕ Índice





Usos do Documento de Requisitos

- ⊕ Planejamento
- ⊕ Projeto
- ⊕ Implementação
- ⊕ Teste
- ⊕ Gerenciamento de Mudanças
- ⊕ Uso do sistema e manutenção
- ⊕ Gerenciamento do Contrato





Critérios de Qualidade para Requisitos

Os critérios de qualidade definidos na norma IEEE 830-1998 podem ser aplicados tanto para requisitos individuais como para documentos completos de requisitos



- ⊕ Acordado
- ⊕ Priorizado
- ⊕ Não-ambíguo
- ⊕ Válido e atualizado
- ⊕ Correto
- ⊕ Consistente
- ⊕ Verificável
- ⊕ Realizável
- ⊕ Rastreável
- ⊕ Completo
- ⊕ Compreensível





Regras fundamentais para a Legibilidade de Requisitos

- ⊕ Frases Curtas e parágrafos curtos.
- ⊕ Formular apenas um requisito por frase.





Glossário

- ⊕ Uma causa frequente de conflitos na Engenharia de Requisitos ocorre quando pessoas envolvidas no processo de desenvolvimento de Software têm diferentes interpretações de termos.
- ⊕ Um glossário é uma coleção de definição de termos, apresentando os seguintes elementos:
 - ✓ Termos técnicos do contexto
 - ✓ Abreviações e acrônimos
 - ✓ Conceitos do dia-a-dia
 - ✓ Sinônimos
 - ✓ Homônimos





Negociação de Requisitos

- ⊕ As melhores negociações buscam ao máximo um resultado “ganha-a-ganha”.
- ⊕ A arte da Negociação:
 - ✓ Ambas partes têm de ter a sensação de que ganharam ou atingiram algum objetivo;
 - ✓ Planejar uma estratégia;
 - ✓ Ouvir atentamente;
 - ✓ Concentrar-se no interesse da outra parte. Posições rígidas geram conflitos;
 - ✓ Não deixar a negociação ir para o lado pessoal;
 - ✓ Ser criativo;
 - ✓ Estar preparado para se comprometer.



Fonte: Pressman



Validação de Requisitos

✦ Uma revisão do modelo de requisitos trata das seguintes questões:

- ✓ Cada um dos requisitos é consistente com os objetivos globais do software ?
- ✓ Todos os requisitos foram especificados no nível adequado de abstração ?
- ✓ O requisito é realmente necessário (essencial) ?
- ✓ Cada requisito é apresentado sem ambiguidade ?
- ✓ Cada um dos requisitos possui uma atribuição ? (Há algum usuário relacionado ?)
- ✓ Algum requisito conflita com outros requisitos ?
- ✓ Cada um dos requisitos é atingível no nível técnico ?
- ✓ Cada um dos requisitos pode ser testado, uma vez implementado ?
- ✓ O modelo de requisitos reflete, de forma apropriada, as funções do sistema ?
- ✓ O modelo de requisitos foi estruturado (“dividido”) de modo adequado ?



Fonte: Pressman