

MBA em Engenharia de Software a Distância

**Disciplina:
Análise e Gerência de Requisitos**



Definindo o Sucesso do Software

- Clientes satisfeitos
- Eles estão satisfeitos quando você:
 - **Atende às expectativas**
 - **Entrega no prazo**
 - **Entrega no orçamento**



0 **Sucesso começa com a Gerência de Requisitos**



Principais Fatores de Falha dos Projetos

- ✓ Falta de “Input” do Usuário
- ✓ Objetivos não estavam claros
- ✓ Falta de suporte do nível executivo
- ✓ Ignorar um grupo de clientes
- ✓ Requisitos e especificações incompletos
- ✓ Requisitos e especificações instáveis (mudanças)
- ✓ Omitir um grupo de requisitos
- ✓ Permitir inconsistências entre grupos de requisitos
- ✓ Aceitar requisito inadequado, incorreto, indefinido, ou impreciso
- ✓ Aceitar um requisito ambíguo e inconsistente



Como os Projetos podem ter sucesso?

- Análise do Problema
 - Entenda o problema
 - Obtenha concordância dos envolvidos
- Levantamento dos Requisitos
 - Identifique quem usará o sistema (atores)
 - Descubra como o sistema será usado (casos de uso)
- Gerência de Requisitos
 - Especifique os requisitos completamente
 - Gerencie expectativas, mudanças e erros
 - Controle o aumento do escopo
 - Defina a equipe e a mantenha informada



Engenharia de requisitos

- Porque usar?

- Construir uma solução “**bonitinha**”, mas que não resolve o problema a que se propõe, não é interessante.
- Saber o que o cliente precisa é de extrema importância para o sucesso de qualquer proposta de solução.
- Usando as **técnicas e ferramentas da engenharia de requisitos** podemos ter uma idéia bem próxima do impacto que a solução a ser proposta terá sobre o negócio, sobre as expectativas do cliente e como será a interação entre usuário final e o produto do nosso trabalho.



Engenharia de requisitos

- O processo:
 - Com a engenharia de requisitos começamos nosso trabalho **definindo o escopo e a natureza do problema** a ser resolvido (concepção), passamos para definição do que é necessário para a solução do problema (levantamento dos requisitos), **melhoramos os requisitos que foram definidos** (elaboração), durante o levantamento e a elaboração são definidas as prioridades e a importância dos requisitos levantados (negociação), e finalmente o problema é especificado, revisado ou validado para garantir que o entendimento é o mesmo para todos os envolvidos.



Engenharia de requisitos

- O produto
 - Ao término da análise de requisitos, teremos um ou vários documentos que permitirão à equipe de desenvolvedores e clientes terem o mesmo entendimento do produto a ser desenvolvido.



Requisitos

- Os requisitos de um sistema de computação constituem uma **especificação** das características e propriedades do sistema.
- É uma descrição do que o sistema **deve fazer**, de **como** ele deve se comportar, bem como das suas restrições de operação.



Requisitos

- Requisito é uma condição ou capacidade necessária para um usuário resolver um problema ou alcançar um objetivo; para satisfazer uma especificação em um sistema ou em um componente; com uma representação documentada.

*The IEEE Standard Glossary
of Software Engineering Terminology, 1997.*

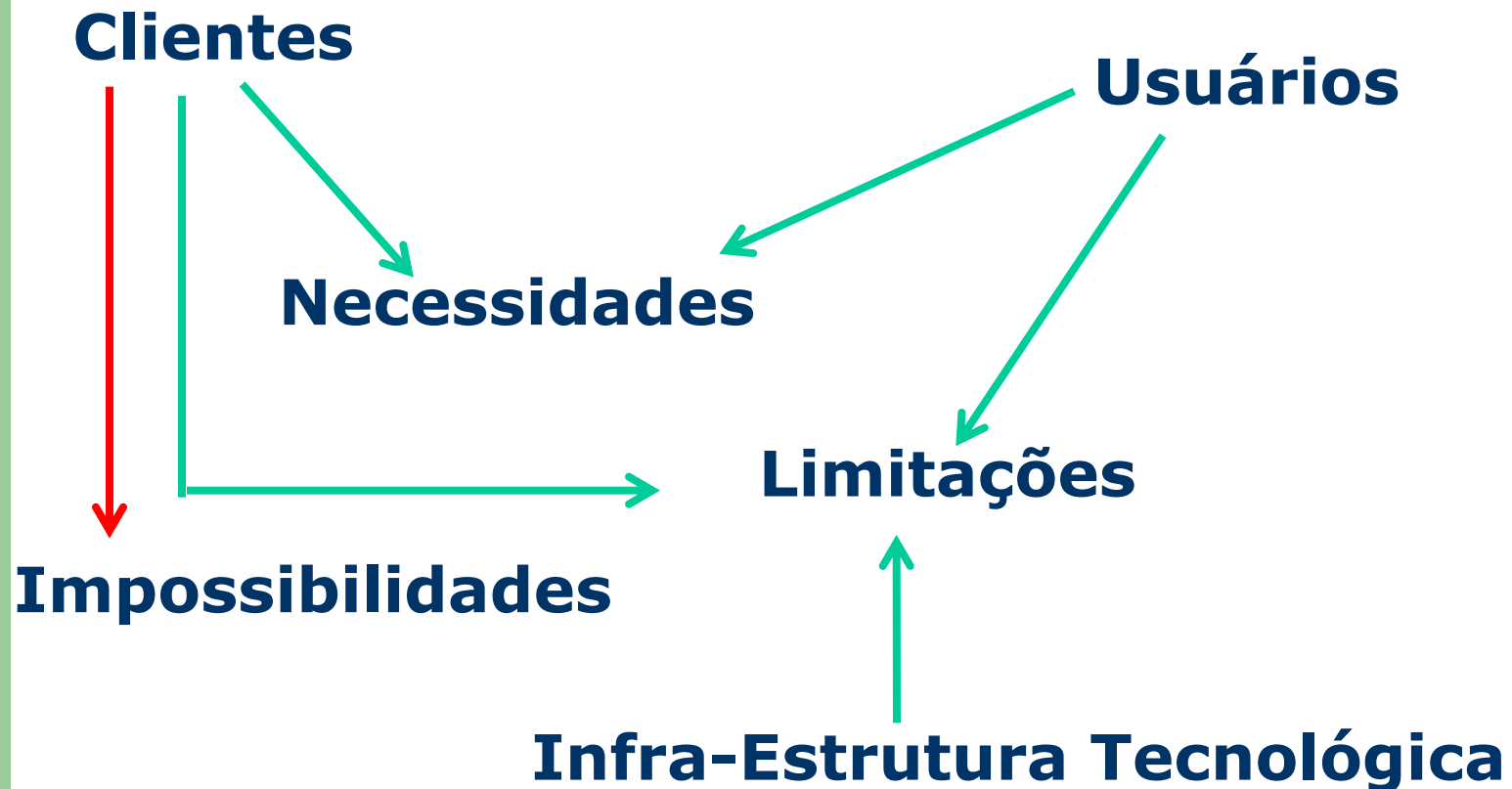


Requisitos

- É importante ressaltar que os requisitos descrevem "o que o sistema deve fazer"- e também "o que ele não deve fazer"- sem dizer "o como fazer".
- Quando o requisito é expresso em termos do comportamento do sistema, este comportamento deve ser possível de ser percebido por um observador externo ao sistema.



Requisitos



Análise de Requisitos

Trecho do Pequeno Príncipe: Antoine Saint-Exupéry, 1996.

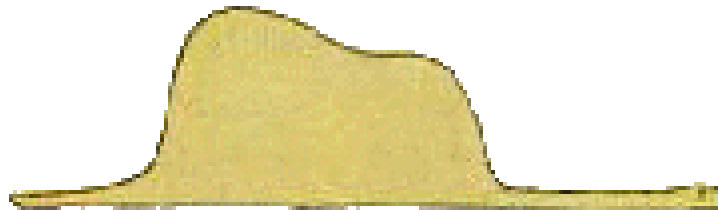
“E ele repetiu-me então, brandamente, como uma coisa muito séria:

- Por favor ... desenha-me um carneiro ...

Quando o mistério é muito impressionante, a gente não ousa desobedecer. Por mais absurdo que aquilo me parecesse a mil milhas de todos os lugares habitados e em perigo de morte, tirei do bolso uma folha de papel e uma caneta. Mas lembrei-me, então, que eu havia estudado de preferência geografia, história, cálculo e gramática, e disse ao garoto (com um pouco de mau humor) que eu não sabia desenhar. Respondeu-me:

-Não tem importância. Desenha-me um carneiro.

Como jamais houvesse desenhado um carneiro, refiz para ele um dos dois únicos desenhos que sabia. O da jibóia fechada. E fiquei estupefato de ouvir o garoto replicar:



Análise de Requisitos

- Não! Não! Eu não quero um elefante numa jibóia. A jibóia é perigosa e o elefante toma muito espaço. Tudo é pequeno onde eu moro. Preciso é de um carneiro. Desenha-me um carneiro.

Então eu desenhei.



Olhou atentamente, e disse:

- Não! Esse já está muito doente. Desenha outro.

Desenhei de novo.

-Bem vês que isto não é um carneiro. É um bode... Olha os chifres...

-Fiz mais uma vez o desenho.

Mas ele foi recusado como os precedentes:

- Este aí é muito velho. Quero um carneiro que viva muito.

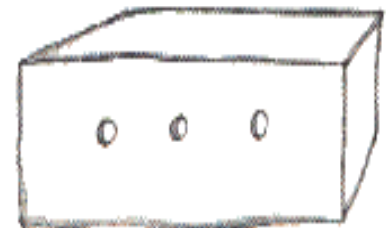
-Então, perdendo a paciência, como tinha pressa de desmontar o motor, rabisquei o desenho ao lado.

E arrisquei:

-Esta é a caixa. O carneiro está dentro.

Mas fiquei surpreso de ver iluminar-se a face do meu pequeno juiz:

- Era assim mesmo que eu queria! Será preciso muito capim para esse carneiro?"



Análise de Requisitos

Gerenciar Requisitos não é fácil

- Requisitos:
 - Nem sempre são óbvios
 - Chegam por várias fontes
 - Nem sempre são facilmente expressos em palavras
 - Estão relacionados entre si e entre outros produtos do processo de engenharia de software
 - Possuem propriedades e valores únicos
 - MUDAM!!
- Um grande **número de requisitos se torna não gerenciável**, caso não controlado.



Requisitos e Especificação

- **Requisito (IEEE)**
 - Uma condição ou **capacidade necessitada** por um usuário para resolver um problema ou alcançar um objetivo.
 - Uma condição ou capacidade que deve ser satisfeita por um sistema para **satisfazer um contrato** ou um padrão.
- **Especificação:**
 - descrição rigorosa e minuciosa das características que um material, uma obra, **ou um serviço** deverá apresentar.
 - processo de **representação dos requisitos** de uma forma que leva à implementação bem-sucedida.



Requisitos e Especificação

- **Importância da Especificação Correta**
 - Uma compreensão completa dos Requisitos do Software é fundamental para obter um software e um processo de desenvolvimento com alta qualidade.
 - Não importa quão bem projetado ou codificado está um programa, se ele for **mal analisado e especificado** desapontará o usuário e trará aborrecimentos ao desenvolvedor



Requisitos e Especificação

- **Importância da Especificação Correta**
 - Ser a base para o desenvolvimento;
 - Permitir o controle da qualidade do produto;
 - Estabelecer a comunicação entre o pessoal envolvido no projeto;
 - Auxiliar no entendimento do problema.



Requisitos e Especificação

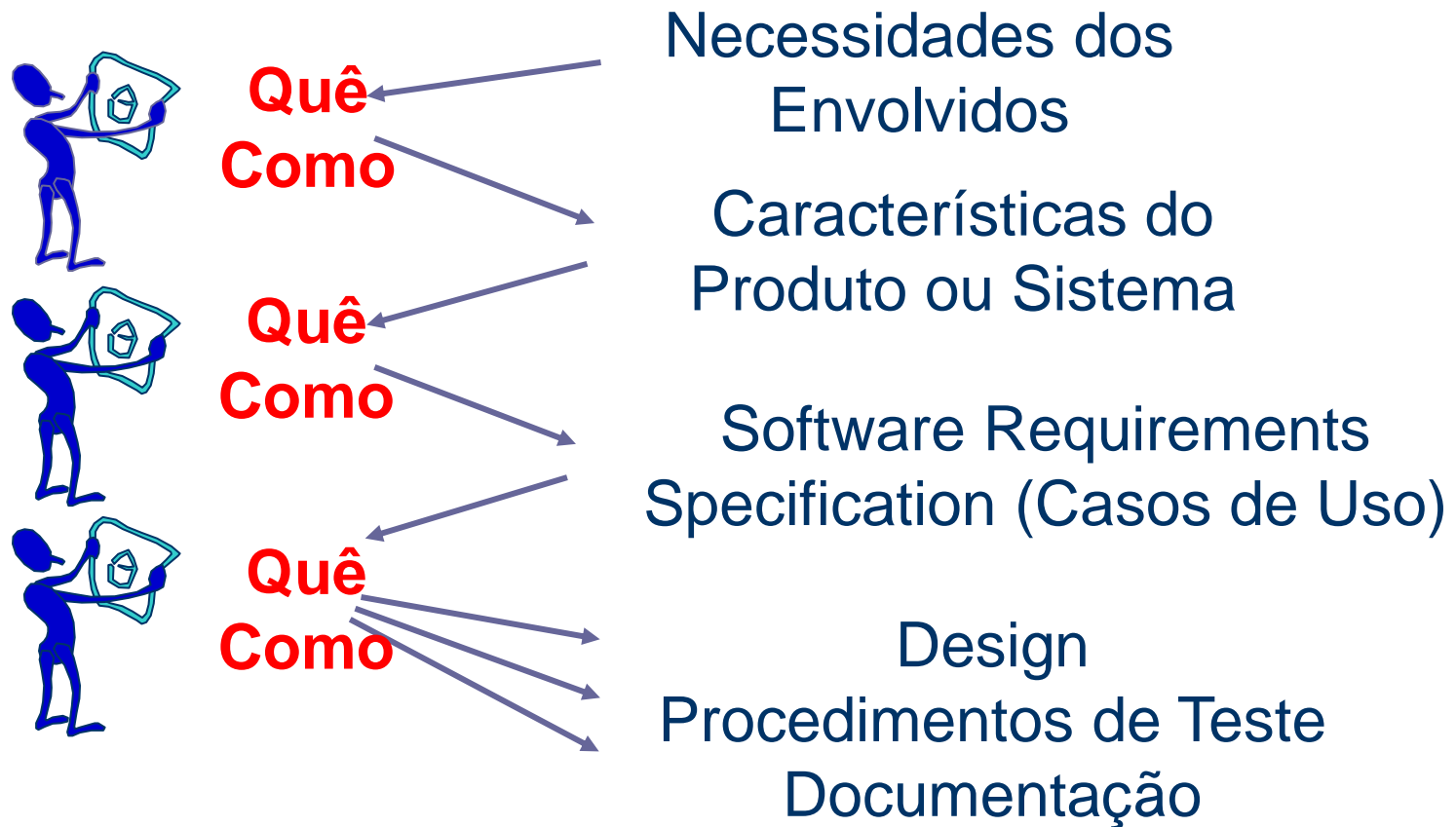
Princípios

- Entender o problema antes de começar a produzir modelos;
- Desenvolver **protótipos que auxiliem** ao usuário visualizar como seus requisitos podem ser atendidos;
- Registrar a origem e a justificativa de cada requisito;
- Utilizar múltiplas visões dos requisitos ;
- Priorizar requisitos;
- Trabalhar para eliminar ambigüidades.



Análise de Requisitos

Requisitos existem em diversos níveis



Engenharia de Requisitos de Software

- “Nosso mundo é um lugar no qual as pessoas não sabem aquilo **que querem** e estão dispostas a passar pelo inferno para consegui-lo.” (Don Marques).
- O inferno dos requisitos é aquele círculo particular do inferno em que **Sisyphus** está empurrando uma pedra montanha acima, somente para vê-la rolar de volta para baixo.
 - Frequentemente mal interpretado, esse mito possui raízes na realidade. Sisyphus, na verdade, alcançou o topo da montanha muitas vezes; só que ficava se perguntando se já havia terminado, **sendo forçado a recomeçar**.



Engenharia de Requisitos de Software

- Engenharia de Requisitos é um amplo campo de pesquisa para a **Engenharia de Software**.
- Como área específica de atuação, apresenta avanços tecnológicos e meios próprios voltados para pesquisa em:
 - ***Terminologia, métodos, linguagens e ferramentas que compõem o esforço de pôr em prática o campo de conhecimento gerado.***



Engenharia de Requisitos de Software

- ER: Termo relativamente novo que foi inventado para cobrir todas as atividades envolvidas em:
 - descobrimento, documentação e manutenção de um conjunto de requisitos para um sistema baseado em computador.
- Pode ser considerado como o processo através do qual o documento de requisitos é preenchido.

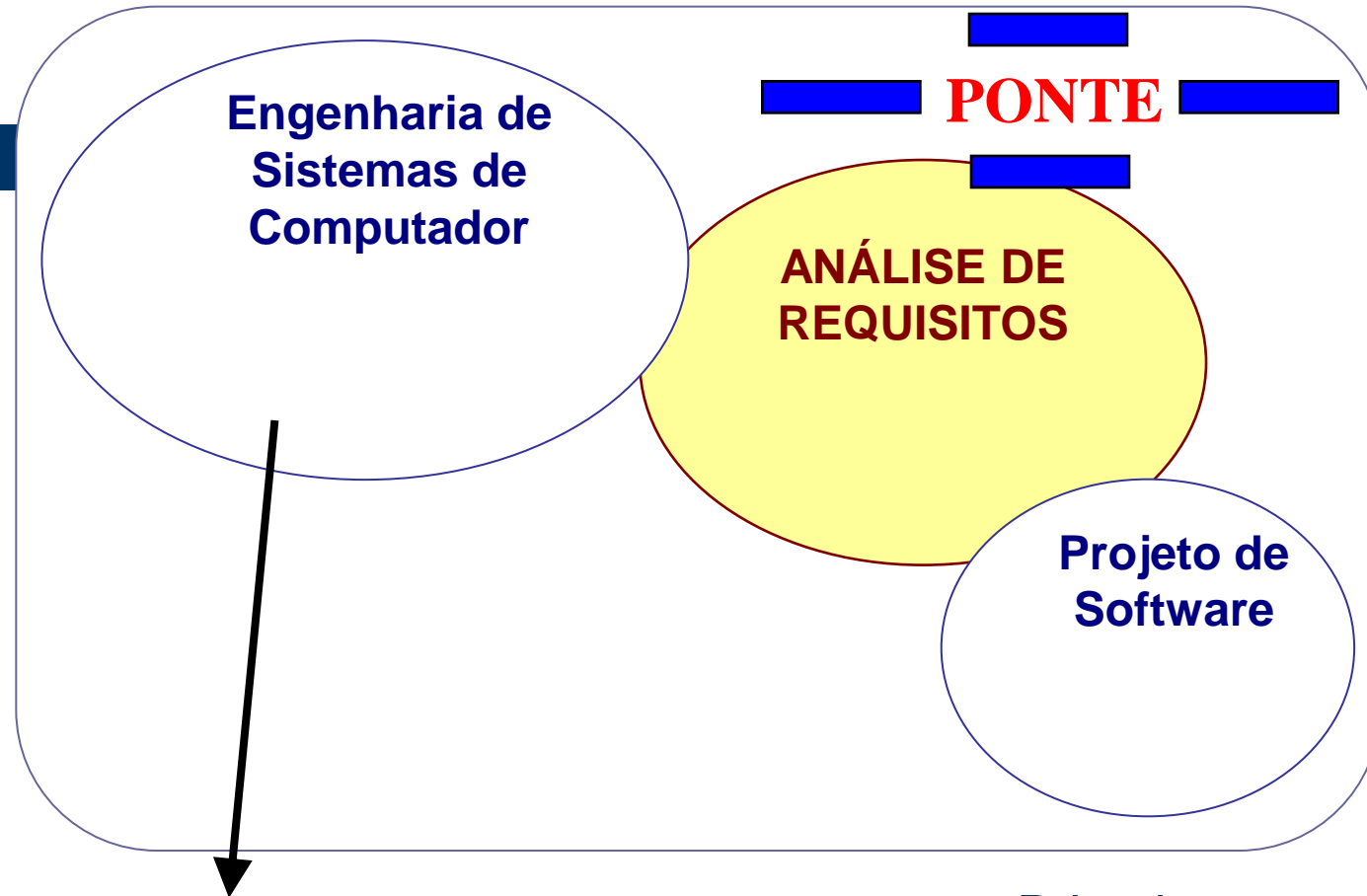


Engenharia de Requisitos de Software

- A engenharia de requisitos tem por objetivo **estabelecer os serviços que devem** ser fornecidos pelo sistema e as **restrições** sob as quais ele deve operar.



Engenharia de Requisitos de Software – Fase de Análise de Requisitos

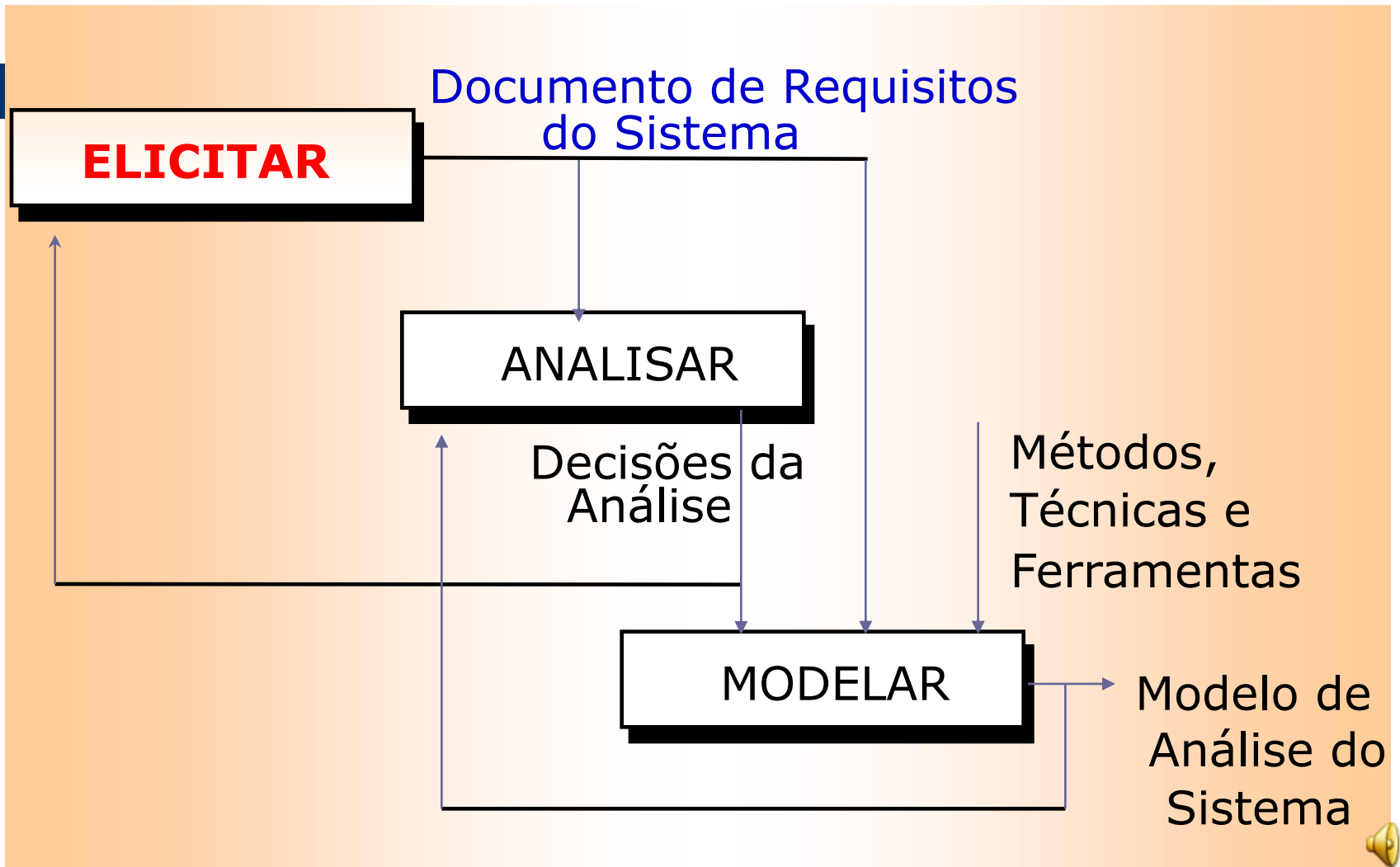


Escopo do Software

- Primeiro passo técnico
- Analista de Sistemas



Engenharia de Requisitos de Software – Principais fases da Engenharia de Requisitos



Engenharia de Requisitos de Software – Principais fases da Engenharia de Requisitos

Elicitação dos requisitos

- Nesta fase o engenheiro de requisitos procura **captar os requisitos** do software, buscando obter conhecimento do domínio do problema.
- **ELICITAR:** descobrir, tornar explícito, obter o máximo de informações para o conhecimento do objeto em questão.
- Cabe à elicitação a tarefa de identificar os fatos relacionados aos requisitos do Sistema, de forma a prover o mais correto e mais completo entendimento do que é demandado do software.
- Para alcançar tal objetivo, esta fase utiliza três atividades principais: identificação das fontes de informação; **coleta de fatos e comunicação, além de ferramentas, pessoal e métodos.**



Engenharia de Requisitos de Software – Principais fases da Engenharia de Requisitos

Elicitação dos Requisitos



- Obter informação sobre domínio do problema e sistema atual (Antes de manter as reuniões com os clientes e usuários e identificar os requisitos, é fundamental **conhecer o domínio do problema** e os contextos organizacional e operacional (situação atual). A equipe responsável pelo levantamento deve se familiarizar com o vocabulário próprio do domínio a ser considerado.
- Preparar e realizar reuniões de levantamento /negociações (Utilizar técnicas específicas para o **levantamento de requisitos e técnicas de negociação**).
- Identificar e revisar os objetivos do sistema (Identificar e revisar quais informações relevantes para o cliente que o sistema deverá gerir e armazenar.)
- Identificar e revisar os requisitos funcionais
- Identificar e revisar os requisitos não funcionais



Engenharia de Requisitos de Software – Principais fases da Engenharia de Requisitos

Necessidades da Elicitação

- *Faz* Coleta de Fatos
- *Faz* Identificação de Fontes de Informação
- *Faz* Comunicação
- *Faz/Usa* Ferramentas
- *Usa* Pessoal
- *Usa* Métodos
- *Depende de* Pontos de Vista



100%



O Processo de Engenharia de Requisitos

- **Estudo de Viabilidade:** Analisa se as necessidades do cliente podem ser satisfeitas usando as **tecnologias de software e hardware disponíveis** e se o projeto pode ir adiante, dadas as restrições orçamentárias existentes.
- **Análise de Requisitos:** Descobre os requisitos do sistema através da observação de sistemas existentes, da discussão com usuários e da análise da tarefa.
 - Gera um **conjunto de modelos** que ajudam o analista a compreender o sistema que será especificado.



O Processo de Engenharia de Requisitos

- **Definição dos Requisitos:** Transforma a informação obtida durante a etapa de análise em um documento que define um conjunto de requisitos realmente desejados pelo cliente.
 - O documento deve ser compreendido pelo usuário final e pelo cliente do sistema.
- **Especificação dos Requisitos:** Os requisitos são detalhados e descritos de forma precisa.
 - Serve como base de contrato entre o cliente e o desenvolvedor.



O desafio do levantamento de requisitos

- Determinar e gerenciar **requisitos**, embora seja uma das mais **importantes atividades** do processo de desenvolvimento é, muitas vezes, tratada com **pouco** ou até nenhum esmero.
- Determinar os requisitos é, em última análise, entender exatamente **o que** deve ser feito e **o que** se espera receber como resultado. Neste ponto ainda não se questiona **como** o trabalho será realizado.



As Features de um Sistema

- A **coleta de requisitos** faz parte de todo projeto, e as técnicas se aplicam a qualquer sistema.
- **Ambigüidade:** A menos que você goste do vaivém de usuários e programadores irritados, a meta ao **coletar requisitos** é reduzir a ambigüidade até o ponto em que você possa produzir um projeto útil, que faça o que as pessoas querem.



As Features de um Sistema

- é um estado em que você encontra **interpretações múltiplas** e **contraditórias** em um texto ou situação.
- Você cria **ambigüidades** quando declara os requisitos para seu sistema que alguém possa interpretar de maneiras diferentes.



As Features de um Sistema

- **Persistência:** Persistir aqui significa **duas** coisas: continuar rolando aquela pedra e descobrir o que os **requisitos** significam para seus dados persistentes.
 - Você pode passar sua vida inteira coletando requisitos em vez de desenvolver sistemas.
 - Saber quando persistir ao **reunir e interpretar requisitos** e quando ir adiante vem com a **experiência**. Você pode utilizar medidas sofisticadas para julgar a adequação de seus requisitos, ou pode usar sua intuição.



As Features de um Sistema

- **Acertando suas prioridades:** Coletar os requisitos não é o fim de seu trabalho. Nem todo requisito possui a mesma importância dos outros.
 - A tarefa seguinte é **priorizar** os requisitos compreendendo-os, relacionando-os uns aos outros e categorizando-os.
 - Até você chegar a um nível razoável de detalhes, pode haver dificuldade para perceber quais são as verdadeiras prioridades e que coisas dependem de outras coisas.



As Features de um Sistema

- **Compreendendo os requisitos:** O fato de você possuir uma lista do que as pessoas esperam que seu sistema faça, não significa que você **compreende os requisitos**.
 - A lógica do erro afirma que todo sistema resiste aos esforços dos seres humanos em mudá-lo.



As Features de um Sistema

Considere as seguintes recomendações da **psicologia cognitiva:**

- Declare metas de maneira clara.
- Compreenda onde você deve acomodar metas contraditórias.
- Estabeleça prioridades e, então, modifique-as quando necessário.
- Modele o sistema, incluindo os efeitos colaterais e modificações a longo prazo.
- Compreenda de quantos dados você precisa e consiga-os.



As Features de um Sistema

- Não seja excessivamente abstrato.
- Não reduza tudo a um resumo, uma única causa ou fato.
- Escute os outros quando eles lhe disserem que você está indo pelo caminho errado.
- Aplique métodos quando for útil e evite-os quando eles atrapalharem.
- Estude sua história e aplique as lições aprendidas ao novo sistema.
- Pense em termos de sistemas, não em termos de simples requisitos.



As Features de um Sistema

- **Objetivos Operacionais:** Os objetivos operacionais expressam o propósito existente por trás do sistema e a abordagem **que você pretende assumir** para alcançar esse propósito.
 - Metas, funções, comportamentos, operações são todos sinônimos de objetivos operacionais.
- **Objetivos:** São o tipo mais importante de requisito, porque eles expressam o significado por trás do sistema, o **porquê** do **que** você está fazendo.



As Features de um Sistema

- **Propriedades de objetos:** Quando começar a desenvolver os requisitos de dados para o sistema, começará a ver os objetos (entidades, tabelas, classes, e assim por diante) que existirão no sistema.
 - Propriedades de objetos são a **segunda categoria de requisito**, e provavelmente a mais importante para o projetista do banco de dados.
 - As propriedades podem ser colunas de tabelas, atributos de objetos ou membros de dados de classe.



As Features de um Sistema

- **Regras:** São requisitos condicionais sobre **propriedades de objetos**. Para o valor de uma propriedade ser aceitável, ele deve satisfazer a condição.
 - As regras de **regulação de tempo** também se enquadram nesta categoria. Por exemplo, você pode possuir um objetivo operacional de que o sistema responda uma solicitação de **informações dentro de dois segundos**.



As Features de um Sistema

- Uma outra categoria de regra **é a regra de qualidade**. Pode-se especificar tolerâncias de erro que se tornem parte dos requisitos.
 - Por exemplo, armazenar evidências de DNA, erros de identificação da pessoa individual devem ser menores do que 0,0001, o que significa que os resultados são estatisticamente significativos, com probabilidade 0,9999.



As Features de um Sistema

- **Preferências:** É uma condição sobre um objeto que expressa um estado preferido. Por exemplo, um detetive preferiria conhecer o criminoso por trás de cada crime. De maneira realista, você deve ter histórias de casos não solucionados no sistema - histórias de casos que não tenham criminosos.



As Features de um Sistema

- **Relacionando os requisitos:** Os requisitos não são pérolas de conhecimento isoladas que existem independentemente uns dos outros.
 - O próximo passo na **compreensão de requisitos** é tratá-los como um sistema, **relacionando-os uns aos outros**. A combinação sinérgica de requisitos geralmente pode mudar a natureza da sua abordagem do projeto.



As Features de um Sistema

- **Priorizando os requisitos** : Os requisitos assumem muitos aspectos. Alguns são vitais para o projeto; outros, são menos importantes.
 - Precisamos passar por muitos requisitos não substanciais para treinar a mente a **distinguir** entre algo não substancial e a coisa verdadeira.
 - Devem-se priorizar os **objetivos operacionais** por sua contribuição à missão do sistema . Se o sucesso do sistema exige o alcance bem sucedido do objetivo, **este é crítico**.



Características dos Requisitos

- **Verificáveis:**

- Se não podemos verificar a satisfação de um dado requisito tanto faz ele existir ou não. A verificação ocorre através de procedimentos de teste, experimentos e provas ou através de acordos de aceitação previamente definidos

- **Precisos:**

- Requisitos devem ser expressos precisamente, de outro modo não se pode garantir que irão ser interpretados da mesma forma por todas as pessoas envolvidas.



Características dos Requisitos

- **Corretos**
 - Requisitos devem expressar corretamente o que é requerido.
- **Consistentes**
 - Requisitos não devem conter conflitos.
- **Completo**
 - Tudo que é requerido deve ser expresso.



Características dos Requisitos

- **Compreensíveis**

- Todas as pessoas envolvidas devem entender, no seu nível de participação, o que está expresso em um requisito.

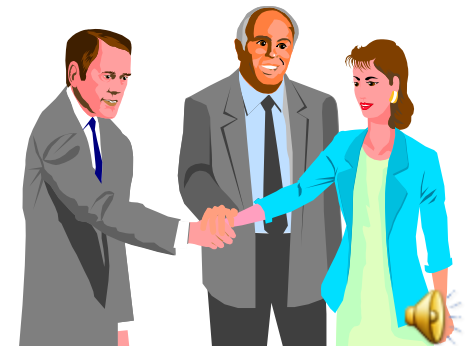
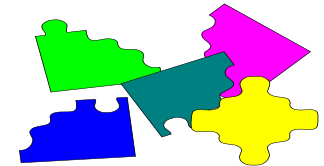
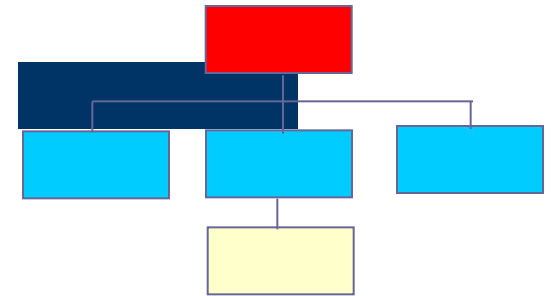
- **Manuteníveis**

- Devemos poder mudar a expressão de um requisito quando este muda.



Estratégia de levantamento de dados e de requisitos

- 1. Proceder à **pesquisa institucional da empresa** e do **órgão** a ser estudado (leitura de estatuto, regimento interno, organograma, normas, regulamentos e outros documentos), identificando a sua localização (endereço);
- 2. Posicionar o **órgão (departamento)** na estrutura da empresa, identificando os seus relacionamentos com os demais órgãos internos e externos (relacionar a troca de informações);
- 3. Identificar as **chefias** (do órgão e dos órgãos hierarquicamente superior a ele) e expor sucintamente o trabalho que será desenvolvido, **solicitando a participação** delas, pois o comprometimento do usuário é muito importante em todas as fases.



Técnicas de Levantamento de Requisitos

- Entrevistas
- Workshop
- BrainStorm
- Teatralização
- Joint Application Design (JAD) - IBM



Técnicas de Levantamento de Requisitos - Entrevistas

- **Entrevistas:** A entrevista é um instrumento fundamental que pode ter uma grande **variedade de objetivos**, como no caso do jornalista, chefe de empresa, diretor de escola, professor, juiz, etc. Aqui nos interessa a entrevista, entendida como aquela na qual se busca a investigação dos requisitos.



Técnicas de Levantamento de Requisitos - Entrevistas

- Baseia em uma técnica com seus **procedimentos** ou regras empíricas com os quais não só se amplia e se verifica como também, se aplica o conhecimento **científico**.
 - A **técnica é o ponto de interação** entre a ciência e as necessidades práticas; é assim que a entrevista alcança a aplicação de conhecimentos científicos e, ao mesmo tempo, obtém ou possibilita levar a vida diária do ser humano ao nível do conhecimento e da elaboração científica.



Técnicas de Levantamento de Requisitos - Entrevistas

- A entrevista é um instrumento muito **difundido** e devemos delimitar o seu alcance em função de suas regras ou indicações práticas de sua execução.
- Pode ser de dois tipos fundamentais: **aberta e fechada**.
 - Na **fechada** as perguntas já estão previstas, assim como a ordem e a maneira de formulá-las, e o entrevistador não pode alterar nenhuma destas disposições.



Técnicas de Levantamento de Requisitos - Entrevistas

- Na **aberta**, o entrevistador tem uma liberdade para as perguntas ou para suas intervenções, permitindo-se toda a flexibilidade necessária em cada caso particular.
- A entrevista **fechada** é, na realidade, um **questionário** que passa a ter uma relação estreita com a entrevista, na medida em que uma manipulação de certos princípios e regras facilita e possibilita a aplicação do questionário.



Técnicas de Levantamento de Requisitos - Entrevistas

- A **aberta** não se caracteriza essencialmente pela liberdade de **colocar perguntas**, pois há uma flexibilidade suficiente para permitir, na medida do possível, que o entrevistado **configure o campo da entrevista** segundo sua estrutura psicológica particular, ou de outra maneira – que o campo da entrevista se configure, o máximo possível, pelas variáveis que **dependem da personalidade** do entrevistado.



Técnicas de Levantamento de Requisitos - Entrevistas

- A realidade é que, em todos os casos, a entrevista é sempre um fenômeno **grupal**, já que mesmo com a participação de um só entrevistado sua relação com o entrevistador deve ser considerada.
- Podem-se diferenciar também as entrevistas segundo o beneficiário do resultado:
 - a) entrevista que se realiza em **benefício do entrevistado** - que é o caso do **levantamento de requisitos**;



Técnicas de Levantamento de Requisitos - Entrevistas

- b) a entrevista cujo objetivo é a **pesquisa**, na qual importam os resultados científicos;
 - c) a entrevista que se realiza para um **terceiro** (uma instituição).
- Um ponto fundamental é que o entrevistador **desperte interesse e participação**, que “**motive**” o entrevistado.



Técnicas de Levantamento de Requisitos - Entrevistas

- A **chave fundamental** da entrevista está na investigação que se realiza durante o seu transcurso. As observações são sempre registradas em função das hipóteses que o observador vai emitindo.
- Uma **técnica de entrevista** direta de pessoa para pessoa que seja eficiente exige a preparação de uma lista de **perguntas elaboradas** para se obter uma compreensão dos problemas reais e das possíveis soluções.



Técnicas de Levantamento de Requisitos - Entrevistas

- Para obter respostas o mais imparcial possível, é preciso **certificar-se** de que as perguntas feitas serão sem contexto. As perguntas **sem contexto** são perguntas abstratas de alto nível que podem ser feitas na **fase inicial de um projeto** para obter informações sobre propriedades globais dos problemas **dos usuários e de suas possíveis soluções.**



Técnicas de Levantamento de Requisitos - Entrevistas

- Uma pergunta sem contexto é:
 - Sempre apropriada.
 - Formulada para ajudá-lo a entender as perspectivas dos envolvidos.
 - Imparcial em relação ao conhecimento das soluções ou à sua opinião de quais seriam as soluções.



Técnicas de Levantamento de Requisitos - Entrevistas

- Abaixo são relacionadas algumas **dicas**:
 - Faça uma pesquisa prévia da formação dos envolvidos ou dos usuários e da empresa.
 - Revise as perguntas antes da entrevista.
 - Consulte o formato durante a entrevista para assegurar que as perguntas certas estão sendo feitas.
 - Resuma os dois ou três problemas principais no final da entrevista. Repita o que você aprendeu para confirmar sua compreensão.



Técnicas de Levantamento de Requisitos - Entrevistas

- Exemplos de perguntas sem contexto usadas para identificar os usuários :
 - Quem é o cliente?
 - Quem é o usuário?
 - Suas necessidades são diferentes?
 - Quais são suas formações, habilidades, ambientes?



Técnicas de Levantamento de Requisitos - Entrevistas

- Exemplos de perguntas sem contexto que ajudam a **compreender** os processos de negócios:
 - Qual é o problema?
 - Qual é a razão para que se deseje resolver esse problema?
 - Existem outras razões para que se deseje resolver esse problema?
 - Qual é a importância de uma solução bem-sucedida?



Técnicas de Levantamento de Requisitos - Entrevistas

- Como se resolve o problema agora?
- Qual é o ponto de equilíbrio entre tempo e valor?
- Em que outro lugar a solução para esse problema pode ser encontrada?



Técnicas de Levantamento de Requisitos - Entrevistas

- Exemplos de **perguntas sem contexto** que ajudam a compreender os requisitos do sistema ou do produto a ser criado:
 - Que problema esse produto resolve?
 - Que problemas de negócios esse produto poderá ocasionar?
 - Que riscos poderão existir para o usuário?
 - Que ambiente o produto encontrará?



Técnicas de Levantamento de Requisitos - Entrevistas

- Quais são as suas expectativas em relação à usabilidade?
- Quais são as suas expectativas em relação à confiabilidade?
- Que desempenho/precisão é exigido?
- Exemplos de meta perguntas sem contexto:
 - Estou fazendo muitas perguntas?
 - Minhas perguntas parecem relevantes?
 - Você é a pessoa certa para responder a essas perguntas?



Técnicas de Levantamento de Requisitos - Entrevistas

- As suas respostas são requisitos?
- Posso fazer mais perguntas depois?
- Você aceitaria participar de uma revisão de requisitos?
- Há algo mais que eu poderia perguntar a você?
- Ao formular um conjunto de perguntas, devemos levar em consideração o seguinte:
 - Não peça às pessoas que descrevam algo que não costuma descrever.



Técnicas de Levantamento de Requisitos - Entrevistas

- Não faça perguntas que pressuponham que os usuários podem descrever atividades complexas.
- Em geral, as pessoas podem fazer muitas atividades que não conseguem descrever.
- Evidências empíricas - poucas correlações.
- Faça perguntas para as quais não haja respostas definitivas.
- Evite perguntas iniciadas por "Por que...?", já que poderão levar a uma postura defensiva.

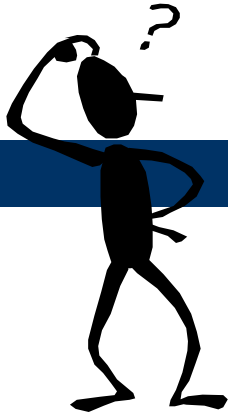


Técnicas de Levantamento de Requisitos - Entrevistas

- Ao conduzir uma sessão de entrevista, lembre-se:
 - Não espere respostas simples.
 - Não apresse o entrevistado em suas respostas.
 - Ouça, ouça, ouça!



Técnica do 5 W / 2 H:



QUEM (WHO)?

Quem faz?
Quem está fazendo?
Quem deveria estar fazendo?
Quem mais pode fazer?
Quem mais deveria fazer?

O QUE (WHAT)?

O que fazer?
O que está sendo feito?
O que deveria ser feito?
O que mais pode ser feito?
O que mais deveria ser feito?

ONDE (WHERE)?

Onde fazer?
Onde está sendo feito?
Onde deveria ser feito?
Onde mais pode ser feito?
Onde mais deveria ser feito?

COMO (HOW)?

Como fazer?

Como é feito?

Como deveria ser feito?

Este método pode ser usado em outras áreas?

Existe outra maneira de fazer?

POR QUE (WHY)?

Porque fazer?
Porque é feito?
Porque fazer aqui?
Porque fazer agora?
Porque fazer assim?

QUANDO (WHEN)?

Quando fazer?
Quando é feito?
Quando deveria estar fazendo?
Quem mais pode fazer?
Quem mais deveria fazer?

QUANTO (HOW MUCH)?

Quanto custa?
Quanto será o investimento inicial?
Qual o custo final do projeto?



Técnicas de Levantamento de Requisitos - Workshops

- **Workshops: Preparação do Workshop:**
Conduzir um workshop de requisitos implica **reunir todos** os envolvidos durante um período intensivo, concentrado.
 - Um Analista de Sistemas **atua** como um facilitador da reunião. Todos os participantes deverão **contribuir** ativamente e os resultados da sessão deverão ser disponibilizados imediatamente para eles.



Técnicas de Levantamento de Requisitos - Workshops

- O **workshop de requisitos** fornece um framework para aplicar as técnicas de identificação, por exemplo, **brainstorming**, **encenação**, **interpretação de papéis** e **revisão dos requisitos existentes**. Essas técnicas poderão ser usadas isoladamente ou combinadas.
 - Todas poderão ser combinadas com as funcionalidades do sistema. Podemos criar uma ou algumas **encenações** para cada funcionalidade prevista para o sistema.



Técnicas de Levantamento de Requisitos - Workshops

- Podemos usar a interpretação de papéis como uma **maneira de compreender** como os usuários usarão o sistema e para ajudá-los a definir as funcionalidades.
- O facilitador de um **workshop de requisitos** precisará estar preparado para as seguintes dificuldades:
 - É possível que os envolvidos saibam o que desejam, mas não consigam expressar isso.
 - É possível que os envolvidos não saibam o que desejam.



Técnicas de Levantamento de Requisitos - Workshops

- Os envolvidos poderão achar **que sabem o que desejam** até que você lhes dê o que eles disseram que desejavam.
- Os analistas poderão achar que **compreendem** os problemas dos usuários melhor do que eles próprios.
- As pessoas poderão achar que as demais pessoas estão politicamente motivadas.



Técnicas de Levantamento de Requisitos - Workshops

- **Antes do Workshop**

- O facilitador precisará "**vender**" o workshop para os envolvidos que deverão estar presentes e estabelecer o grupo que participará dele.
- Os participantes deverão receber material de estudo de "**aquecimento**" antes do workshop. O facilitador é o **responsável pelas atividades**: enviar convites, encontrar um local adequado com o equipamento necessário e distribuir uma agenda.



Técnicas de Levantamento de Requisitos - Workshops

- **Conduzir a Sessão**

O facilitador presidirá a sessão, o que inclui:

- Dar a todos a oportunidade de falar.
- Manter a sessão sob controle.
- Reunir informações para Requisitos aplicáveis.
- Registrar as descobertas.
- Resumir a sessão e elaborar conclusões.



Técnicas de Levantamento de Requisitos - Workshops

- **Consolidar Resultados:** Após o **workshop de requisitos**, o facilitador (juntamente com os colegas analistas de sistemas) precisará dedicar um tempo à síntese das descobertas e ao resumo das informações.
- **Truques do Negócio:** Um conjunto de problemas e as soluções **sugeridas** que poderão atender **às necessidades** do facilitador. As soluções referem-se a um conjunto de "**cartões**" que podem parecer desnecessários, mas que, se mostram muito eficientes.



Técnicas de Levantamento de Requisitos - Workshops

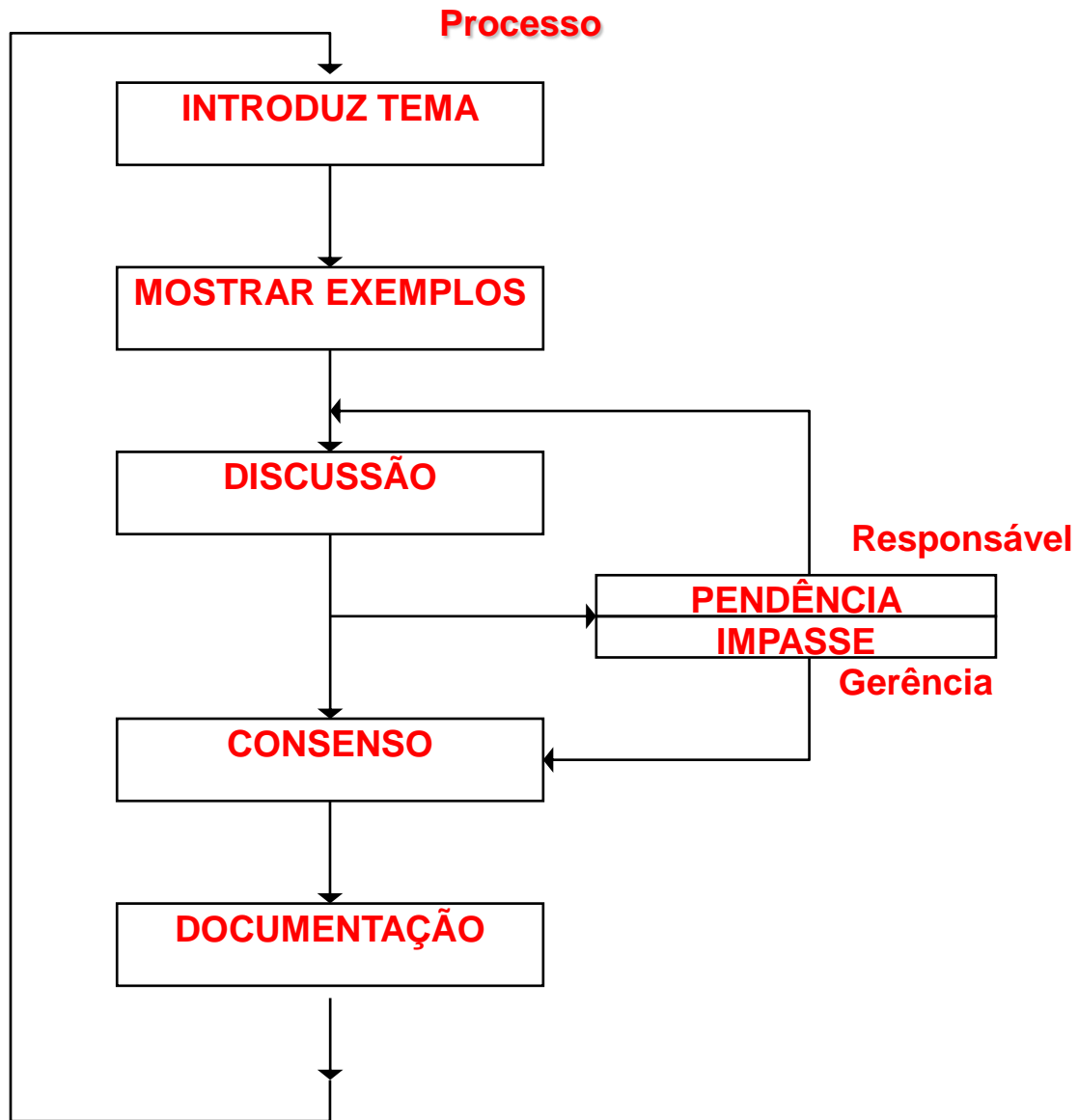
Problema	Solução
Dificuldade de retomar o workshop após os intervalos.	Qualquer pessoa atrasada leva um cartão "Atrasado". Use um timer de cozinha para chamar a atenção das pessoas, utilize uma caixa semelhante às utilizadas nas arrecadações de caridade (estabeleça R\$1,00 para cada cartão usado).
Críticas severas—preconceitos mesquinhos, disputas de território, politicagens e comentários vulgares.	Cartão "1 Comentário Vulgar Grátis", cartão "É uma Excelente Idéia!!" .
Exibicionismos, pontos de vista autoritários, informações irregulares dos participantes.	Use um facilitador treinado, limite o tempo do discurso a uma "Expressão de Opinião de Cinco Minutos".
Baixa de rendimento após o almoço.	Sirva almoços, lanches, café, refrigerantes, balas e biscoitos leves, reorganize a sala, altere a temperatura.



Técnicas de Levantamento de Requisitos - JAD

- **JAD - Joint Application Development:** Usuários e desenvolvedores **trabalham juntos** em uma reunião com o objetivo de:
 - identificar o problema;
 - propor elementos de solução;
 - negociar diferentes abordagens;
 - especificar um conjunto preliminar de requisitos de solução.
- **Envolve:**
 - preparação para reunião a partir de uma requisição geral do produto.
 - Reunião.





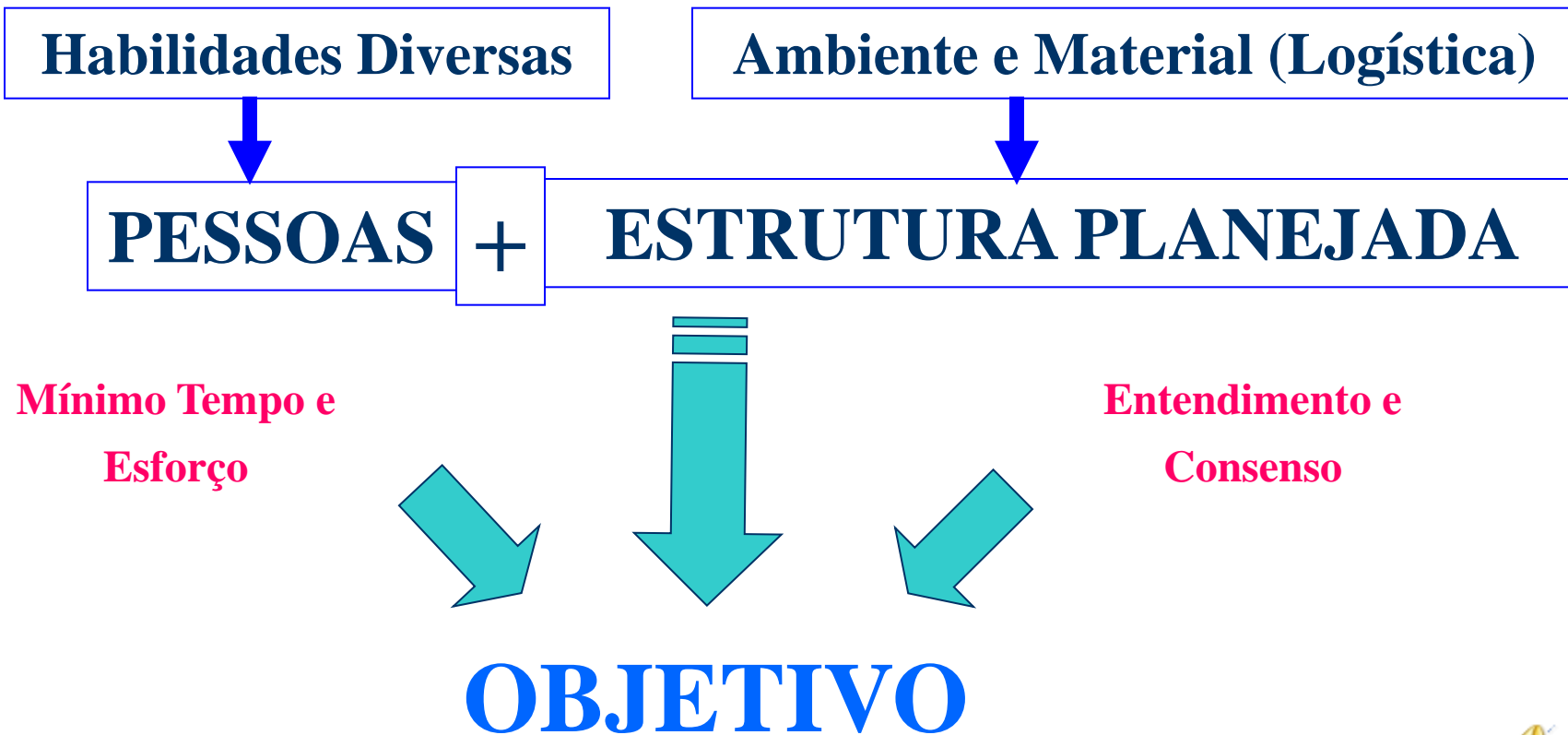
Técnicas de Levantamento de Requisitos - JAD

Definição

- Técnica que reúne determinado número de pessoas em sessões bem estruturadas para, com tempo e esforço reduzidos, consolidar um objetivo pré-determinado.



Técnicas de Levantamento de Requisitos - JAD



Técnicas de Levantamento de Requisitos - JAD

Histórico

- Dificuldade da solução das diferenças de ponto de vista entre usuários.
- Buscando a **obtenção de consenso** entre um grande grupo de pessoas.
- A metodologia JAD foi desenvolvida pela IBM, no Canadá, em 1977.



Técnicas de Levantamento de Requisitos - JAD

Metodologia: Alguns princípios para o JAD:

- Comprometimento.
- Envolvimento de todas as pessoas necessárias para atingir o objetivo;
- Líder experiente e imparcial;
- Representante da alta gerência;
- Sessões com no máximo 15 pessoas;
- Planejamento logístico;
- Decisões pelo consenso;
- Documentação dos resultados produzidos;



Técnicas de Levantamento de Requisitos - JAD

Componentes da equipe do JAD (são as estrelas do espetáculo)

- Facilitador;
- Patrocinador;
- Especialistas da Informação;
- Documentador;
- Representantes dos usuários/clientes;
- Outros convidados.



Técnicas de Levantamento de Requisitos - JAD

Organização do JAD

- Reunião Inicial;
- Reunião de Revisão;
- Reuniões para a Realização das Sessões do Projeto;

Reunião Inicial

- Primeira fase dos trabalhos.
- Reunião com duração aproximada de 3 h, participação das pessoas do nível estratégico.
- Definem os objetivos, participantes, agenda e distribuição de tarefas.



Técnicas de Levantamento de Requisitos - JAD

Organização do JAD

Reunião de Revisão

- Dura normalmente 3 h, com os mesmos participantes da reunião inicial.
- Revisão de todos os pontos combinados na reunião inicial.
- Correções e ajustes dos problemas detectados.
- Verificação do cumprimento de tarefas.



Técnicas de Levantamento de Requisitos - JAD

Organização do JAD

Reuniões das Sessões de Projeto (formato)

- São reuniões orientadas pelo líder/facilitador.
- Usa dinâmica de grupo para gerar idéias.
- Os analistas de informações ajudam a controlar e registrar os trabalhos.
- Participação de todos os relacionados no plano para o JAD (definidos nas reuniões inicial e de revisão).



Técnicas de Levantamento de Requisitos - JAD

Organização do JAD

Reuniões das Sessões de Projeto (Sequência)

- Abertura dos trabalhos;
- Apresentação e distribuição do organograma da empresa.
- Recordação do fluxo atual.
- Problemas, defeitos e empecilhos.
- Definição dos objetivos.
- Racionalização do fluxo atual.



Técnicas de Levantamento de Requisitos - JAD

Organização do JAD

Reuniões das Sessões de Projeto (Sequência)

- Proposição do novo fluxo.
- Cronograma, prazos e responsabilidades.
- Encerramento.



Técnicas de Levantamento de Requisitos - JAD

Implementação do JAD

Recomendado uma Abordagem em 4 etapas :

- Execução de um projeto piloto.
- Determinação da natureza do grupo de suporte.
- Incorporação do JAD.
- Ampliação do uso do JAD.



Técnicas de Levantamento de Requisitos - JAD

Conclusão

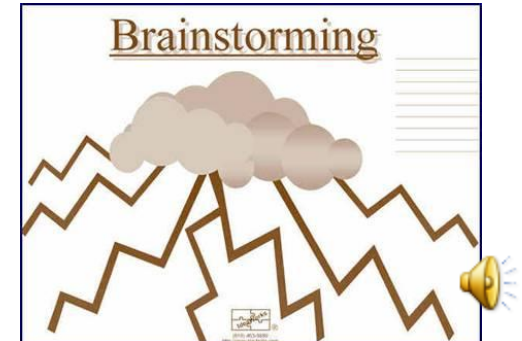
A eficiência comprovada com o uso do JAD proporciona os seguintes benefícios:

- Acelera o processo na obtenção dos resultados.
- Solução de conflitos com rapidez.
- Espírito de equipe.
- Compromisso dos participantes com o projeto.



Técnicas de Levantamento de Requisitos - Brainstorm

- **Brainstorm ou Tempestade Cerebral:**
Freqüentemente a tempestade cerebral permite um **desbloqueio**, um aquecimento para a **conversação** entre o analista e o cliente.
 - Tem como propostas o desenvolvimento da **originalidade e da desinibição**, bem como a produção de um grande número de **idéias** em prazo curto; numa palavra, à **criatividade**.



Técnicas de Levantamento de Requisitos - Brainstorm

- Seu funcionamento geral é o seguinte:
 - dado um tema, o cliente(s) expressa oralmente, em uma **palavra** ou em **frases bem curtas**, tudo o que lhe vem à cabeça.
 - O analista vai anotando tudo e a seguir faz a **seleção das idéias**, agrupando-as por alguma semelhança.



Técnicas de Levantamento de Requisitos - Brainstorm

- A atividade de **brainstorming** implica que todas as pessoas do **workshop** se dediquem durante um curto período de tempo, a expressar tudo o que acham importante para o projeto.
- **As regras de brainstorming** são as seguintes:
 - Começar definindo claramente o objetivo da sessão de brainstorming.
 - Gerar o maior número possível de idéias.



Técnicas de Levantamento de Requisitos - Brainstorm

- Deixar a imaginação fluir.
- Não permitir críticas ou debates durante a reunião de informações.
- Depois da reunião das informações, transformar e combinar as idéias.



Técnicas de Levantamento de Requisitos - Teatralização

- **Teatralização:** Filmes, desenhos animados, recursos animados, todos começam com **encenações** que informam quem são usuários, **o que** acontece a eles e **como** acontece.
 - Ajuda a reunir e **restringir os requisitos** dos clientes de uma maneira amigável para os usuários.
 - Estimula soluções de **design** mais criativas e inovadoras.



Técnicas de Levantamento de Requisitos - Teatralização

- Estimulam a **revisão em equipe** e evita os recursos indesejáveis.
- Assegura que os **recursos** sejam implementados de maneira intuitiva e acessível.
- Facilita o **processo de entrevista** - evitando a síndrome da página em branco.



Técnicas de Levantamento de Requisitos - Teatralização

- De uma maneira simples, a encenação implica usar uma ferramenta para ilustrar (e algumas vezes animar) para os usuários como o sistema se ajustará à organização e também indicar como ele se comportará.
 - Um facilitador mostra uma encenação inicial para o grupo e este último faz comentários.
 - A encenação se desenrola em "tempo real" durante o workshop.



Técnicas de Levantamento de Requisitos - Teatralização

- Há dois grupos distintos de ferramentas a serem usados na encenação: **ferramentas passivas** e **ferramentas ativas**. As ferramentas **passivas** indicam que serão mostradas figuras não animadas, enquanto as ferramentas **ativas** têm recursos internos mais sofisticados.



Técnicas de Levantamento de Requisitos - Teatralização

- São exemplos de ferramentas **passivas** de encenação:
 - Papel e lápis;
 - Blocos de anotações autocolantes;
 - Construtores GUI;
 - Diferentes tipos de gerenciadores de apresentação.



Técnicas de Levantamento de Requisitos - Teatralização

- São exemplos de ferramentas **ativas** de encenação:
 - HyperCard, SuperCard;
 - Demo-It(tm) II da Bricklin;
 - Macromedia Director e outras ferramentas de animação;
 - PowerPoint.



Técnicas de Levantamento de Requisitos - Teatralização

- Avisos e comentários:
 - É necessário que as **encenações** possam ser criadas e alteradas com facilidade. Se você não alterou nada é sinal de que não aprendeu nada.
 - Não faça uma **encenação** muito boa. Não é nem um protótipo nem uma demonstração do que é real (percepção realista).



Técnicas de Levantamento de Requisitos - Teatralização

- **Interpretação de Papéis:** Será atribuído a cada membro do grupo um papel de interesse para o sistema. Os papéis são os usuários.
 - O grupo inspecionará então como o sistema é usado. Haverá discussões sobre quem é responsável por o que. Anote as responsabilidades de cada usuário. Fazer com que o analista de sistemas interprete o papel do usuário ou do cliente ajuda a obter um discernimento real do domínio do problema.



Técnicas de Levantamento de Requisitos - Teatralização

- **Revisar Requisitos Existentes:** Podemos consultar especificações de requisitos de sistemas anteriores ou de outros sistemas relacionados - o que poderá ser útil.
 - Com o grupo, inspecione cada requisito para detectar comportamentos de aplicativos ou atributos comportamentais. Em geral, devemos ignorar informações explicativas como introduções e descrições gerais do sistema.



Desenvolvedores

- Usuários não sabem o que querem
- Usuários não podem articular o que eles querem
- Usuários têm muitas necessidades puramente políticas
- Usuários não podem priorizar suas necessidades
- Usuários se recusam a ter responsabilidade pelo sistema
- Usuários não estão comprometidos com o desenvolvimento de projetos
- Usuários não permanecem dentro do planejamento.

Usuários

- Desenvolvedores não entendem necessidades operacionais
- Desenvolvedores colocam muita ênfase em aspectos técnicos
- Desenvolvedores querem definir o que os usuários devem fazer
- Desenvolvedores não conseguem transformar necessidades em um sistema de sucesso
- Desenvolvedores estão sempre atrasados
- Desenvolvedores sempre querem tempo e esforço
- Desenvolvedores são incapazes de responder rapidamente sobre necessidades de modificação.



MBA em Engenharia de Software a Distância

**Disciplina:
Análise e Gerência de Requisitos**



O Componente Humano (Participantes do Processo)

- O desenvolvimento de software é uma tarefa **altamente cooperativa**. Tecnologias complexas demandam **especialistas em áreas específicas**.
- Uma equipe de desenvolvimento de sistemas de software pode **envolver vários especialistas**:
 - **Profissionais de informática** para fornecer o conhecimento técnico necessário ao desenvolvimento do sistema de software e,
 - **Especialistas do domínio** para o qual o sistema de software deve ser desenvolvido.



O Componente Humano (Participantes do Processo)

- Uma equipe de desenvolvimento de software consiste de um:
 - Gerente;
 - Analistas;
 - Arquiteto de Software
 - Projetistas;
 - Programadores;
 - clientes e;
 - grupos de avaliação de qualidade.



O Componente Humano (Participantes do Processo)

- A descrição dos participantes do processo tem um fim didático.
 - Na prática, **a mesma pessoa** desempenha diferentes funções e, por outro lado, **uma mesma função** é normalmente desempenhada por **várias pessoas**.



O Componente Humano (Participantes do Processo)

- Em virtude de seu **tamanho e complexidade**, o desenvolvimento de sistemas de software é um **empreendimento realizado em equipe**.
- **Gerentes de projeto:** Como o próprio nome diz, o gerente de projetos é o profissional **responsável pela gerência ou coordenação** das atividades necessárias à construção do sistema.



O Componente Humano (Participantes do Processo)

- Esse profissional também é responsável por fazer:
 - O orçamento do projeto, estimar o tempo necessário para o desenvolvimento do sistema, definir qual o processo de desenvolvimento, o cronograma de execução das atividades, a mão-de-obra especializada, os recursos de hardware e software.
- O acompanhamento das atividades realizadas durante o desenvolvimento do sistema também é tarefa do gerente do projeto



O Componente Humano (Participantes do Processo)

- **Analistas:** O analista de sistemas é o profissional que deve ter conhecimento do *domínio do negócio*. Deve entender os *problemas* do domínio do negócio para que possa definir os *requisitos do sistema* a ser desenvolvido.
 - Devem estar aptos a se comunicar com *especialistas do domínio* para obter conhecimento acerca dos *problemas e das necessidades* envolvidas da organização.



O Componente Humano (Participantes do Processo)

- Deve ter domínio do **vocabulário da área** de conhecimento na qual o sistema será implantado para que, **ao se comunicar** com o especialista de domínio, este não precise ser interrompido a todo o momento para **explicar** conceitos básicos da área.



O Componente Humano (Participantes do Processo)

- O analista de sistemas é o profissional responsável **por entender** as necessidades dos clientes em relação ao sistema a ser desenvolvido e **repassar** esse entendimento aos demais desenvolvedores do sistema.
 - O analista de sistemas representa uma ponte de comunicação entre duas "**facções**": a dos profissionais de computação e a dos profissionais do negócio.



O Componente Humano (Participantes do Processo)

- Com a experiência adquirida através da participação no desenvolvimento de diversos projetos, alguns analistas se tornam gerentes de projetos.
- Outra característica necessária a um analista é a ética profissional.
 - Muitas vezes, o analista de sistemas está em contato com informações sigilosas e estratégicas dentro da organização na qual está trabalhando.



O Componente Humano (Participantes do Processo)

- **Projetistas:** O projetista de sistemas é o componente da equipe de desenvolvimento cujas funções são:
 - (1) **avaliar as alternativas de solução** (da definição) do problema resultante da análise e;
 - (2) gerar a **especificação de uma solução** computacional detalhada. A tarefa do projetista de sistemas é muitas vezes chamada de *projeto físico*.



O Componente Humano (Participantes do Processo)

- Na prática, existem diversos tipos de projetistas:
 - **projetistas de interface** (especializados nos padrões de uma interface gráfica);
 - **projetista de redes** (especializados no projeto de redes de comunicação);
 - **projetista de bancos de dados** (especializados no projeto de bancos de dados) e assim por diante.



O Componente Humano (Participantes do Processo)

- **Arquitetos de software:** Profissional encontrado principalmente em grandes equipes reunidas para desenvolver sistemas complexos. O objetivo desse profissional é elaborar a **arquitetura do sistema** como um todo.
- Além de tomar decisões globais, o **arquiteto** também deve ser capaz de tomar **decisões técnicas** (decisões que têm influência sobre o **desempenho do sistema**).



O Componente Humano (Participantes do Processo)

- Também trabalha em conjunto com o **gerente de projeto** para priorizar e organizar o plano de projeto.



O Componente Humano (Participantes do Processo)

- **Programadores:** Profissional responsável pela implementação do sistema. Normalmente há vários programadores em uma equipe de desenvolvimento.
- Um programador pode ser proficiente em uma ou mais linguagens de programação, além de ter conhecimento sobre bancos de dados.



O Componente Humano (Participantes do Processo)

- A maioria das equipes de desenvolvimento possuem analistas que realizam alguma programação, e programadores que realizam alguma análise.
- O analista de sistemas está envolvido em todas as etapas do desenvolvimento, diferentemente do programador, que participa unicamente das fases finais (implementação e testes).



O Componente Humano (Participantes do Processo)

- **Clientes:** O cliente é o indivíduo, ou grupo de indivíduos, para o qual o sistema é construído.
- Podem-se distinguir dois tipos de clientes: o **cliente usuário e o cliente contratante**.
 - O cliente usuário é o indivíduo que efetivamente utilizará o sistema. Normalmente é um especialista no **domínio do negócio**. É com esse tipo de cliente que o analista de sistemas interage para levantar os **requisitos do sistema**.



O Componente Humano (Participantes do Processo)

- O **cliente contratante** é quem solicita o desenvolvimento do sistema. Usuário que encomenda e patrocina os custos de desenvolvimento e manutenção.
- Em pequenas organizações, o cliente usuário e o cliente contratante **são a mesma pessoa**. Em grandes organizações, o cliente contratante faz **parte da gerência e é responsável por tomadas de decisões estratégicas** dentro da empresa, enquanto que o cliente usuário é o **responsável pela realização de processos operacionais**.



O Componente Humano (Participantes do Processo)

- A única maneira de se ter um usuário **satisfeito** é torná-lo um legítimo participante no desenvolvimento do sistema.
- Não importa qual seja o processo de desenvolvimento utilizado; o envolvimento do **usuário no desenvolvimento** de um sistema de software é de fundamental importância.



O Componente Humano (Participantes do Processo)

Questão para reflexão:

- Você é que participante do Processo?



MBA em Engenharia de Software a Distância

**Disciplina:
Análise e Gerência de Requisitos**



O Processo de Requisitos de Software

- **Requisitos de software:** Os problemas que os engenheiros de software têm para solucionar são, muitas vezes, **imensamente complexos**.
 - Compreender a **natureza dos problemas** pode ser muito difícil, especialmente se o sistema for novo. Conseqüentemente, é difícil **estabelecer com exatidão** o que o sistema deve fazer.



O Processo de Requisitos de Software

- As descrições das **funções e das restrições** são os requisitos para o sistema;
- O processo de descobrir, analisar, documentar e verificar essas funções e restrições é chamado de **engenharia de requisitos**.



O Processo de Requisitos de Software

- Um **requisito** é definido como "uma condição ou uma capacidade com a qual o sistema deve estar de acordo".
- “Descrição abstrata dos **serviços** que o sistema deve fornecer e das **restrições** sob as quais o sistema deve operar”



O Processo de Requisitos de Software

- Os requisitos **devem ser redigidos** de modo que os **diversos participantes** possam apresentar propostas, oferecendo diferentes **maneiras de atender** às necessidades organizacionais do cliente.



O Processo de Requisitos de Software

- Uma vez estabelecido um contrato, os participantes precisam preparar uma definição de sistema para o cliente, com mais detalhes, de modo que o cliente compreenda e possa validar o que o software fará.
 - Esse documento pode ser chamado de **documento de requisitos** do sistema.



O Processo de Requisitos de Software

- Tipos de Requisitos
 - Requisitos de negócio (domínio)
 - Requisitos de usuário
 - Requisitos de sistema
 - Requisitos funcionais
 - funcionais (comportamento e propriedade)
 - Requisitos não funcionais
 - não funcionais (qualitativos, quantitativos, restritivos)



O Processo de Requisitos de Software

Requisitos do Usuário

- São declarações, em **linguagem natural** e também em **diagramas**, sobre as **funções** que o sistema deve fornecer e as **restrições** sob as quais deve operar.
- Requisitos de **usuários** descrevem as atividades que os usuários deverão ser capazes de executar com a utilização do sistema.



O Processo de Requisitos de Software

Requisitos do Sistema

- Estabelecem detalhadamente as **funções e as restrições de sistema**.
 - O documento de requisitos de sistema, também chamado de **especificação funcional**, deve ser **preciso**. Serve como um **contrato** entre o **contratante do sistema** e o **desenvolvedor do software**.



O Processo de Requisitos de Software

Requisitos do Sistema

- Exemplo:
 - 1) o software deve oferecer um meio de representar e acessar arquivos externos criados por outras ferramentas
 - 2) o usuário deve dispor de recursos para definir o tipo dos arquivos externos;



O Processo de Requisitos de Software

Requisitos do Sistema

- 3) cada tipo de arquivos externo pode ter uma ferramenta associada que pode ser aplicada a ele;
- 4) cada tipo de arquivo externo pode ser representado com um ícone específico;
- 5) devem ser fornecidos recursos para o ícone que representa um arquivo externo, a ser definido pelo usuário;
- 6) quando um usuário seleciona um ícone que representa um arquivo externo, o efeito dessa seleção é aplicar a ferramenta associada com o tipo de arquivo externo ao arquivo representado pelo ícone selecionado.



O Processo de Requisitos de Software

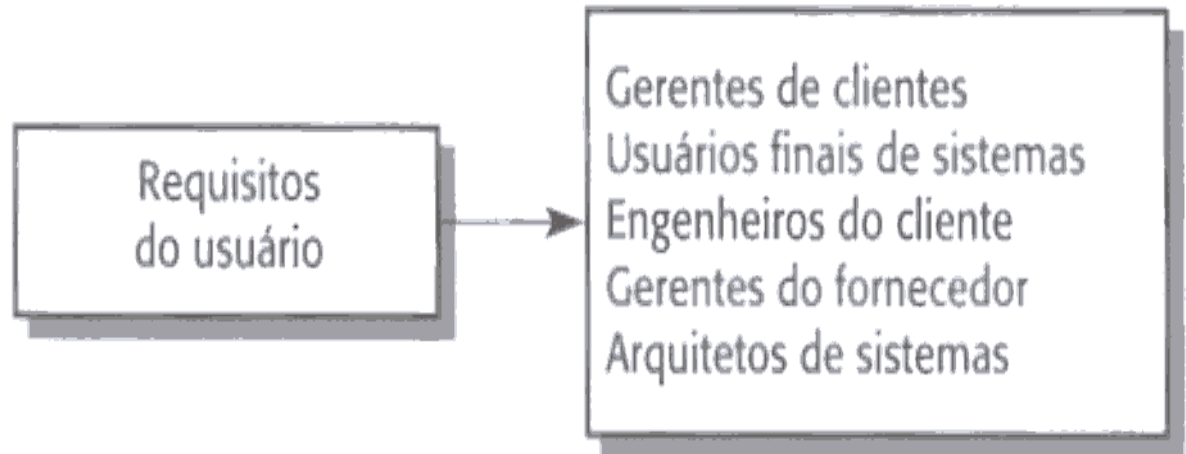
Especificação do Software

- Descrição detalhada do software que serve como base para projeto ou implementação.
 - Escrito para desenvolvedores.



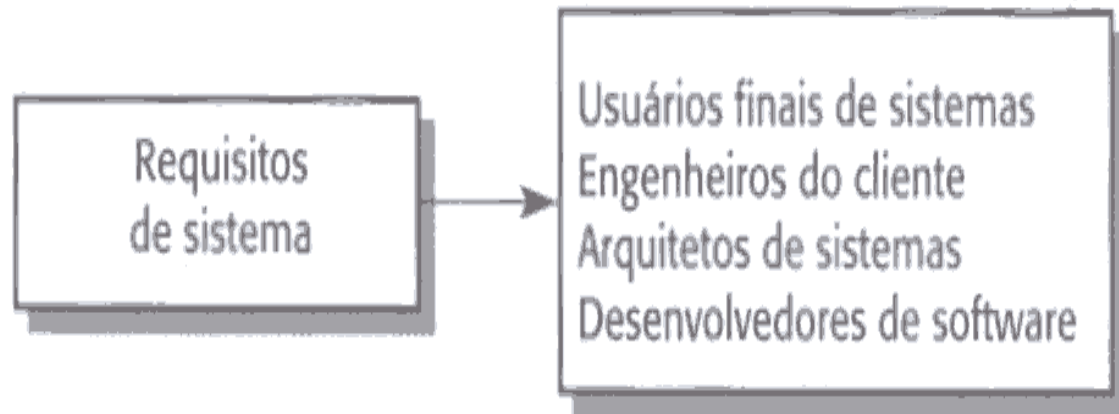
O Processo de Requisitos de Software

- Os requisitos do usuário devem ser escritos para gerentes do cliente e dos participantes, que não tenham um documento técnico detalhado do sistema.



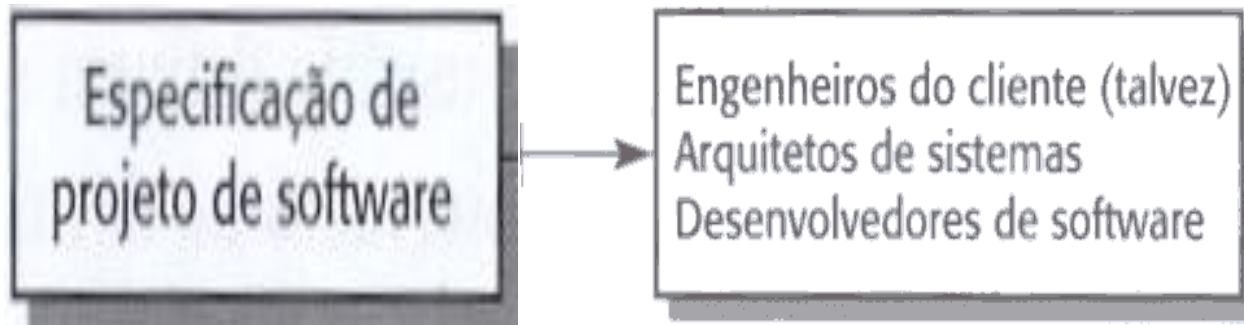
O Processo de Requisitos de Software

- A especificação de requisitos de sistema deve ter como alvo os técnicos e os gerentes de projeto. Será utilizado pelo gerente do cliente e do participante.



O Processo de Requisitos de Software

- A especificação de projeto de software é um documento orientado à implementação. Ele deve ser escrito para os engenheiros de software que desenvolverão o sistema.



O Processo de Requisitos de Software

Requisitos Funcionais e Não-Funcionais

- Os requisitos de sistema de software são, freqüentemente, classificados como:
 - **Funcionais;**
 - **Não-funcionais** ou como **requisitos de domínio** :



O Processo de Requisitos de Software

- **Requisitos Funcionais.** São declarações de funções que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como deve se comportar em determinadas situações.
 - Em alguns casos, os requisitos funcionais podem também explicitamente declarar o que o sistema não deve fazer.



O Processo de Requisitos de Software

- Os **requisitos funcionais** especificam ações que um sistema deve ser **capaz de executar**, sem levar em consideração **restrições físicas**.
- Os **requisitos funcionais** definem as funcionalidades que o sistema deve possuir para que os usuários possam executar suas atividades, de modo a atingir os objetivos do negócio.



O Processo de Requisitos de Software

- **Requisitos Não-funcionais:** São restrições sobre os serviços ou as funções oferecidas pelo sistema.

Entre eles destacam-se **restrições de tempo**, restrições sobre o **processo de desenvolvimento**, **padrões**, entre outros. Exemplo: padrões, regulamento e contratos com os quais o sistema deve ter conformidade; descrição de interfaces externas e requisitos de desempenho;

- **Requisitos de Domínio:** São requisitos que se originam do domínio de aplicação do sistema e que refletem características desse domínio. Podem ser requisitos **funcionais ou não funcionais**.



O Processo de Requisitos de Software

- Os requisitos **de domínio (negócio)** correspondem aos objetivos de negócio - ou do usuário - que devem ser satisfeitos pelo sistema.
- Na realidade, a distinção entre **esses diferentes tipos de requisitos** não é tão clara como sugerem definições simples.
- **Requisitos funcionais:** Para um sistema **descrevem a funcionalidade ou os serviços** que se espera que o sistema forneça. Dependem do tipo de software que está sendo desenvolvido, dos usuários de software que se espera **verificar e do tipo de sistema** que está sendo desenvolvido.



O Processo de Requisitos de Software

- Quando expressos como requisitos de usuário, eles são normalmente descritos de um modo bastante geral, mas os requisitos funcionais de sistema descrevem a função de sistema detalhadamente, suas entradas e saídas, exceções, etc.
- Os requisitos funcionais especificam, o comportamento de entrada e saída de um sistema.
- Requisitos funcionais devem descrever os serviços do sistema em detalhes.



O Processo de Requisitos de Software

- Os **requisitos funcionais** de um sistema podem ser expressos de diversas maneiras. Exemplo: **requisitos funcionais** do sistema de biblioteca para que os estudantes possam pedir livros e documentos de outras bibliotecas:
 - O usuário poderá pesquisar todo o conjunto inicial de **banco de dados** ou **selecionar** um sub-conjunto dele.



O Processo de Requisitos de Software

- O sistema deve oferecer visualizadores apropriados para o usuário ler documentos armazenados.
- A todo pedido deve ser associado um identificador único.



O Processo de Requisitos de Software

- Em princípio, requisitos devem ser completos e consistentes.
- **Completo**
 - Descrições de todos os serviços.
- **Consistência**
 - Não deve haver **conflitos e contradições** nas descrições dos serviços.
- Na prática, torna-se impossível produzir um **documento de requisitos** completo e consistente.



O Processo de Requisitos de Software

- **Requisitos Não Funcionais:** Os requisitos não funcionais, como o nome sugere, são aqueles que não dizem respeito diretamente às funções específicas fornecidas pelo sistema.
 - Podem estar relacionados a propriedades de sistema emergentes, como confiabilidade, tempo de resposta e espaço em disco.



O Processo de Requisitos de Software

- Como alternativa, eles podem definir **restrições** para o sistema, como:
 - a capacidade dos **dispositivos de E/S** (entrada/saída) e;
 - as representações de **dados utilizadas** nas interfaces de sistema.
 - Podem especificar o uso de determinadas **linguagens de programação**, método de desenvolvimento.



O Processo de Requisitos de Software

Requisitos não funcionais compreendem:

- Requisitos de armazenamento de informação;
- Requisitos de produto (ex. rapidez, confiabilidade);
- Requisitos organizacionais (ex. padrões de processo usados);
- Requisitos externos (ex. requisitos legislativos, éticos).
- Requisitos técnicos (desempenho, interface, linguagens de programação, funções de integração etc.);



O Processo de Requisitos de Software

Requisitos não funcionais compreendem:

- Requisitos de interface;
- Requisitos operacionais;
- Requisitos de recursos (potência, memória principal, espaço em disco etc.);
- Requisitos dos testes de aceitação;
- Requisitos de documentação;
- Requisitos de qualidade;
- Requisitos de confiabilidade.



O Processo de Requisitos de Software

Classificação dos Requisitos Não-Funcionais:

- Requisitos do Produto
 - Produto entregue deve comportar-se de **forma particular** (velocidade de execução, confiabilidade, etc). **Exemplo:** Toda consulta ao banco de dados baseada em **código de barras** não deve exceder 5 segundos.



O Processo de Requisitos de Software

Classificação dos Requisitos Não-Funcionais:

- Requisitos Organizacionais
 - Conseqüência de **políticas e procedimentos organizacionais** (padrões de processo usados, requisitos de implementação, etc). **Exemplo:** Todos os documentos entregues devem seguir o **padrão de relatórios XYZ-00**.



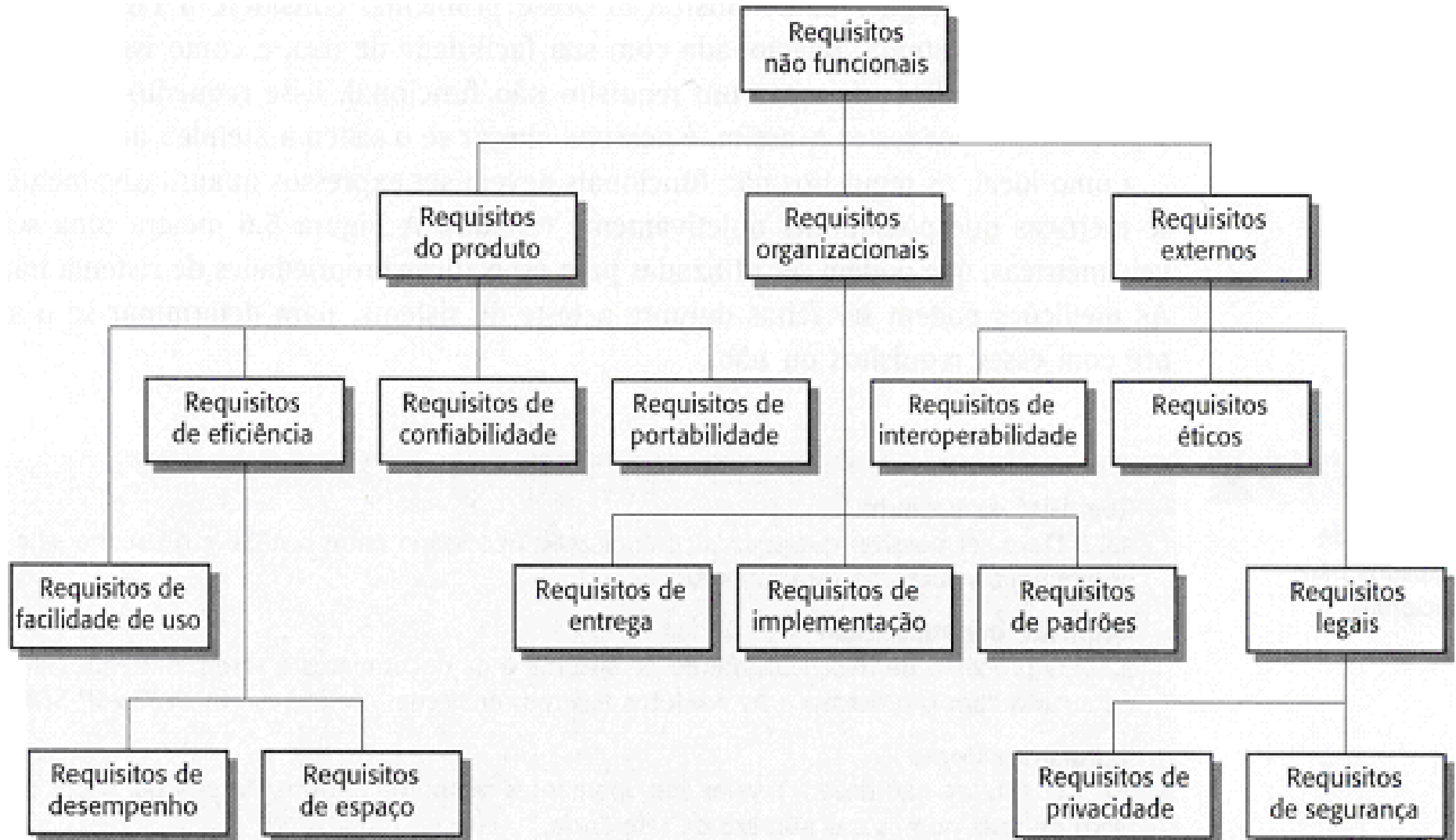
O Processo de Requisitos de Software

Classificação dos Requisitos Não-Funcionais:

- Requisitos Externos
 - Conseqüência de fatores externos ao sistema e ao processo de desenvolvimento (legislação, etc). Exemplo: O sistema não deve usar informações pessoais do usuário com os operadores do sistema.



Tipos de Requisitos Não Funcionais



O Processo de Requisitos de Software

- **Exemplos de Requisitos:**

- O sistema deve prover um formulário para a entrada dos resultados dos testes clínicos de um paciente. (RF)
- A base de dados deve ser protegida para acesso apenas de usuários autorizados. (RNF)
- Dependendo do resultado do teste, somente o Supervisor pode efetuar a entrada do resultado do teste de um paciente. (RNF de confidencialidade).
- O sistema deve emitir um recibo para o cliente, com o tempo máximo de 8 segundos após a transação. (RF “,” RNF de performance).
- O sistema não pode apagar informação de um cliente (?).



O Processo de Requisitos de Software

- **Restrições:** limitam as possibilidades de escolha do desenvolvedor no projeto e na implementação do produto (como por exemplo limitação de recursos e de infra-estrutura); e
- **Atributos de qualidade:** ampliam a descrição das funcionalidades do sistema através da descrição de características de qualidade do produto, que sejam importantes para o cliente e para o desenvolvedor.



O Processo de Requisitos de Software

- **Requisitos de Domínio:** Os requisitos de domínio são derivados do **domínio da aplicação do sistema**, em vez de serem obtidos a partir das necessidades específicas dos usuários do sistema.
 - Eles podem ser **novos requisitos funcionais** em si, podem **restringir os requisitos funcionais existentes** ou estabelecer como devem ser realizados **cálculos específicos**.



Requisitos de Domínio - Exemplos:

- **A desaceleração do trem deve ser computada através da fórmula: $D_{\text{trem}} = D_{\text{controle}} + D_{\text{gradiente}}$.**
- Deve existir uma interface padrão com o usuário para todos os bancos de dados baseada no padrão ABC-98.
- Devido a restrições, 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.



O Processo de Requisitos de Software

- **Requisitos de Usuário:** Os requisitos de usuários devem descrever os requisitos **funcionais e não funcionais** de modo compreensível pelos usuários do sistema que **não têm conhecimentos técnicos** detalhados.
 - Não devem ser definidos utilizando um modelo de implementação. Podem ser **escritos com o uso de linguagem natural, formulários e diagramas.**



O Processo de Requisitos de Software

- Vários problemas podem surgir quando os requisitos são escritos em **linguagem natural**:
 - ***Falta de clareza:*** Às vezes, é difícil utilizar a linguagem de maneira precisa e sem **ambigüidade**, sem produzir um documento de difícil leitura.



O Processo de Requisitos de Software

- ***Confusão de requisitos:*** Os requisitos funcionais e os não funcionais, os objetivos do sistema e as informações sobre o projeto **podem não estar claramente definidos.**
- ***Fusão de requisitos:*** Vários requisitos diferentes **podem ser expressos juntos** como um único requisito.



O Processo de Requisitos de Software

- **Requisitos de Sistema:** São descrições mais detalhadas dos requisitos do usuário.
 - Podem servir como base para um **contrato destinado** à implementação do sistema e, portanto, devem ser uma **especificação completa e consistente** de todo o sistema.
 - São utilizados pelos engenheiros de software como ponto de partida para o **projeto de sistema**.



O Processo de Requisitos de Software

- Uma arquitetura inicial do sistema pode ser definida para ajudar a **estruturar a especificação de requisitos**.
- Na maioria dos casos, os sistemas devem **interoperar** com outros sistemas existentes.
 - Isso restringe o projeto, e essas **restrições geram requisitos** para o novo sistema.



Notações para a especificação de requisitos:

Notação	Descrição
Linguagem natural estruturada	Essa abordagem depende da definição de formulários padrão ou templates para expressar a especificação de requisitos .
Linguagem de descrição de projeto	Essa abordagem utiliza uma linguagem como uma linguagem de programação , mas com recursos mais abstratos para especificar os requisitos pela definição de um modelo operacional do sistema.
Notações gráficas	Uma linguagem gráfica , complementada com anotações de texto , é utilizada para definir os requisitos funcionais do sistema. Mais recentemente, as descrições de use-case foram utilizadas.
Especificações matemáticas	São notações com base em conceitos matemáticos , como uma máquina de estados finitos e conjuntos . Essas especificações não ambíguas reduzem as discussões entre cliente e fornecedor sobre a funcionalidade do sistema . Contudo, a maioria dos clientes não compreende as especificações formais e reluta em aceitá-las no momento de uma contratação de sistema.



O Processo de Requisitos de Software

- **Especificação de requisito em linguagem estruturada:** Destina-se a escrever os requisitos do sistema.
- Quando um formulário-padrão é utilizado para **especificar os requisitos funcionais**, as seguintes informações devem ser incluídas:
 - 1- uma descrição da função ou entidade que está sendo especificada;



O Processo de Requisitos de Software

- 2- uma descrição de suas entradas e de onde elas se originam;
- 3- uma descrição de suas saídas e para onde elas prosseguirão;
- 4- uma indicação de que outras entidades são utilizadas;
- 5- uma descrição dos efeitos colaterais (se existirem) da operação.



Especificações em Linguagem Natural Estruturada - Exemplo

```
Procedure ATM is
  PIN: numPIN;
  conta: numConta;
  saldo: quantia;
  servico: servicosDisponiveis;
  cartaoValido, PINValido: Boolean;
Begin
  loop
    obterCartao(conta, PIN, cartaoValido);
    if cartaoValido then
      validarPIN(PIN, PINValido);
      if PINValido then
        while obterServico(servico) loop
          realizarServico;
        end loop;
      retornaCartao;
    end if;
  end if;
end loop;
End ATM
```



O Processo de Requisitos de Software

- **Especificação de requisitos com o uso de uma PDL:** É possível descrever os requisitos com o uso de uma PDL (*program description language* - linguagem de descrição de programa).
 - Uma PDL é uma linguagem derivada de uma linguagem de programação como **Java** ou **Ada**.



O Processo de Requisitos de Software

- A vantagem de utilizar uma PDL é que ela pode ser verificada **sintática** ou **semanticamente** por ferramentas de software.
- As PDLs resultam em **especificações muito detalhadas** e, algumas vezes, estão muito perto da **implementação** para sua inclusão em um documento de requisitos.



O Processo de Requisitos de Software

- **Especificação de interface:** A grande maioria dos sistemas de software deve operar com outros sistemas que já foram implementados e instalados em um ambiente.
 - Se o novo sistema e os sistemas existentes devem trabalhar em conjunto, as interfaces de sistemas existentes precisam ser especificadas com precisão.



O Processo de Requisitos de Software

- Existem **três diferentes tipos** de interfaces que podem precisar ser definidas:
 - 1) **Interfaces de procedimento**, em que subsistemas existentes oferecem funções, que são **acessadas ao chamar procedimentos** de interface.
 - 2) Estruturas de dados, que são **transmitidas de um subsistema** para outro.
 - 3) **Representações de dados** que foram estabelecidas para um subsistema existente.



PDL baseada em Java de uma interface de servidor de impressão (PrintServer).

```
interface PrintServer {  
    // define um servidor de impressora abstrato  
    // requer: impressora de interface Printer, interface PrintDoc  
    // fornece: initialize, print, displayPrintQueue,  
               cancelPrintJob, switchPrinter  
    void initialize ( Printer p ) ;  
    void print ( Printer p, PrintDoc d ) ;  
    void displayPrintQueue ( Printer p ) ;  
    void cancelPrintJob (Printer p, PrintDoc d) ;  
    void switchPrinter (Printer p 1, Printer p2, PrintDoc d) ;  
} //PrintServer
```



O Processo de Requisitos de Software:

Documento de requisito de software

- Também chamado de **Especificação de Requisitos**, o documento de requisitos é a declaração oficial do que é **requerido** dos desenvolvedores do software. Especifica o que o software **deve fazer** sem dizer **como**.
 - Deve incluir os **requisitos de usuário** para um sistema e uma **especificação detalhada dos requisitos de sistema**. Em alguns casos, os requisitos de usuário e de sistema podem ser integrados em uma única descrição.



O Processo de Requisitos de Software

- O documento deve ser estruturado de forma a facilitar sua manutenção e leitura. Uma estrutura para o **documento de requisitos** pode apresentar as seguintes seções:
 - 1) **Introdução**: Descreve brevemente as funções que o sistema deverá desempenhar.
 - 2) **Glossário**: Define termos técnicos usados no documento.



O Processo de Requisitos de Software

3) **Modelos do Sistema**: Inclui os modelos que mostram o **relacionamento entre os componentes do sistema** e entre o sistema e seu ambiente (modelos de objetos, modelos de fluxo de dados, Modelo Entidade-Relacionamento, etc.).

4) **Definição de Requisitos Funcionais**: Descreve, através de **linguagem natural ou de diagramas**, os serviços que serão fornecidos pelo sistema.



O Processo de Requisitos de Software

- 5) **Definição de Requisitos Não-Funcionais:** Descreve as restrições impostas sobre o software e relaciona estas restrições aos requisitos funcionais.
- 6) **Especificação de Requisitos:** Descreve os requisitos funcionais em mais detalhes.
- 7) ***Requisitos necessários de hardware e software:*** Descreve o hardware e o software de apoio exigidos para realizar o desenvolvimento. Se o hardware tiver de ser comprado, deverão ser incluídos os prazos de entrega e as estimativas de preço.



O Processo de Requisitos de Software

- Revisões são a técnica mais comum de **validação de requisitos**. Nas revisões, usuários e desenvolvedores se **reúnem para discutir o documento de requisitos** em busca de conflitos, contradições, erros e omissões.
- É preciso ter consciência de que os requisitos **sempre mudam** durante o processo de desenvolvimento do software e que o **documento de requisitos** deve ser sempre atualizado face a estas mudanças.



O Processo de Requisitos de Software: Usuários de um documento de requisitos.

Clientes de Sistema	Especificam os requisitos e os lêem para verificar se eles atendem as suas necessidades . Especificam as mudanças nos requisitos.
Gerentes	Utilizam o documento de requisitos para planejar um pedido de proposta para o sistema e para planejar o processo de desenvolvimento do sistema.
Engenheiros de Sistema	Utilizam os requisitos para compreender que sistema deve ser desenvolvido
Engenheiros de Teste de Sistema	Utilizam os requisitos para desenvolver testes de validação para o sistema.
Engenheiros de Manutenção de Sistema	Utilizam os requisitos para ajudar a compreender o sistema e as relações entre suas partes.



AS FASES DO DESENVOLVIMENTO DE SISTEMAS

Como o Sistema
foi descrito no
Levantamento
Inicial



Como
foi
definido
pela
Análise



Como
foi
especificado
no projeto



Como
foi
Implementado



Como foi
corrigido
pela
Manutenção



O que o
usuário
queria!



MBA em Engenharia de Software a Distância

**Disciplina:
Análise e Gerência de Requisitos**



Gerenciamento de Projetos

Objetivos do Módulo

- Proporcionar uma visão geral do gerenciamento de projetos de software. Para podermos:
 - compreender as diferenças entre o **gerenciamento de projetos** de software e outros tipos de gerenciamento de projetos de engenharia;
 - conhecer as principais **tarefas** dos gerentes de projeto de software;
 - compreender por que o **planejamento de projetos** é essencial em todos os projetos de software;



Gerenciamento de Projetos

- compreender como **representações gráficas** (diagramas de barras, diagramas de atividades) são utilizadas pelos gerentes de projeto para representar programações de projeto, e
- compreender o **processo de gerenciamento de riscos** e alguns dos riscos que podem surgir em projetos de software.



Gerenciamento de Projetos

- O Gerenciamento de Projeto de Software é a arte de confrontar os objetivos da concorrência, gerenciar riscos e superar obstáculos para liberar com êxito um produto que atenda às necessidades dos clientes e dos usuários.
 - O fato de que tão poucos projetos sejam indiscutivelmente bem-sucedidos é o comentário suficiente sobre a dificuldade da tarefa.



Gerenciamento de Projetos

- O fracasso de muitos projetos de software, na década de 60 e no início da década de 70, foi a primeira indicação das dificuldades de gerenciamento de software.
 - O software era **entregue com atraso**, não era confiável, custava várias vezes mais do que previam as estimativas originais e, muitas vezes, exibia características precárias de desempenho.
 - Esses projetos não fracassaram porque os gerentes ou os programadores **eram incompetentes**.



Gerenciamento de Projetos

- Precisamos do gerenciamento de projetos de software porque a **engenharia de software** profissional está sempre sujeita às restrições de orçamento e de prazo.
 - Essas **restrições** são estabelecidas pela organização que desenvolve o software.
 - O trabalho do **gerente de projeto** de software é garantir que o projeto de software cumpra **essas restrições** e entregar um produto de software que contribua para as metas da empresa.



Gerenciamento de Projetos

- Os gerentes de software são responsáveis por **planejar e programar** o desenvolvimento do projeto.
 - Supervisionam o trabalho para assegurar que seja realizado em conformidade com os padrões requeridos e monitoram o progresso para verificar se o desenvolvimento está dentro do **prazo e do orçamento**.



Gerenciamento de Projetos

- O bom gerenciamento não pode garantir o sucesso do projeto. Contudo, o mau gerenciamento geralmente resulta no fracasso do projeto:
 - O software é entregue com atraso, custa mais do que originalmente foi estimado e apresenta falha no cumprimento de seus requisitos.



Gerenciamento de Projetos

- A engenharia de software é distinta de outros tipos de engenharia, em uma variedade de modos que podem tornar o gerenciamento de **software particularmente difícil**. Algumas dessas diferenças são:
 - 1) O **produto é intangível**: O gerente do projeto de construção de um navio ou de um projeto de engenharia civil pode ver o produto sendo desenvolvido. Se há um atraso na programação, o efeito no produto é visível.



Gerenciamento de Projetos

Partes da estrutura estão obviamente inacabadas. O **software é intangível**; não pode ser visto ou tocado. Os gerentes de projeto de software não podem **ver** o progresso, eles dependem de outras pessoas para produzir a documentação necessária, a fim de examinar o progresso.

2) **Não há processo de software-padrão** Não temos uma compreensão clara das relações entre o processo de software e os tipos de produto.



Gerenciamento de Projetos

Nas disciplinas de engenharia, o processo é experimentado e testado. Nossa compreensão do **processo de software** se desenvolveu significativamente nos últimos anos. Contudo, ainda não podemos prever com certeza quando um processo de software específico poderá causar problemas de desenvolvimento.



Gerenciamento de Projetos

3) Grandes projetos de software são, freqüentemente, projetos únicos, são normalmente diferentes de projetos anteriores. Os gerentes, na verdade, têm ampla experiência anterior, que pode ser utilizada para reduzir a incerteza nos planos. É difícil prever problemas. As lições aprendidas com experiências podem não ser transferíveis para novos projetos.



Gerenciamento de Projetos

- Por causa desses problemas, não é de surpreender que alguns projetos de software sofram **atrasos e ultrapassem o orçamento** previsto.

Atividades de Gerenciamento

- É impossível dar uma descrição do **trabalho-padrão** de um gerente de software. O trabalho varia muito, dependendo da organização e do produto a ser desenvolvido.



Gerenciamento de Projetos

- A maioria dos gerentes **assume a responsabilidade**, por algumas das seguintes atividades ou por todas elas:
 - 1) elaboração de propostas;
 - 2) planejamento e programação de projetos, custo do projeto;
 - 3) monitoramento e revisões de projetos;
 - 4) seleção e avaliação de pessoal;
 - 5) elaboração de relatórios e apresentações.



Gerenciamento de Projetos

- O primeiro estágio de um **projeto de software** pode envolver a elaboração de uma proposta para executar o projeto.
 - A proposta descreve os objetivos do projeto e como será realizado. Em geral, inclui **estimativas de custo e programação**. Isso pode justificar por que o contrato do projeto deve ser delegado a uma organização ou equipe específica.



Gerenciamento de Projetos

- O planejamento de projeto se ocupa de **identificar as atividades**, os marcos e os documentos a serem produzidos em um projeto.
 - Um plano deve, então, ser traçado para guiar o desenvolvimento em direção aos objetivos do projeto. A **estimativa de custos** é uma atividade que se ocupa de estimar os recursos requeridos para realizar o projeto.



Gerenciamento de Projetos

- O monitoramento de projeto é **uma atividade continua**. O gerente deve manter o acompanhamento do andamento do projeto e comparar os progressos e custos reais com os que foram planejados.
- Durante um projeto, é normal ocorrer uma série de **revisões formais** pelo gerente de projetos. Essas revisões se ocupam em **examinar** o progresso geral, o desenvolvimento técnico do projeto, considerando o **status do projeto** em relação aos objetivos da organização.



Gerenciamento de Projetos

- Os gerentes de projeto, geralmente, precisam selecionar pessoal para trabalhar em seu projeto.
 - O ideal é que uma **equipe hábil** e com **experiência apropriada** esteja disponível para trabalhar no projeto. Contudo, na maioria dos casos, os gerentes têm de se contentar com menos do que uma equipe ideal de projeto. As razões disso são:



Gerenciamento de Projetos

- 1) Orçamento do projeto pode não ser suficiente para contratar uma equipe bem paga, e pode ser necessário utilizar uma **equipe menos** experiente e que exija menor remuneração.
- 2) Uma equipe com a experiência apropriada pode **não estar disponível**, dentro ou fora da organização. Pode ser impossível recrutar uma equipe nova para o projeto. Dentro da organização, as melhores pessoas talvez já **estejam alocadas** em outros projetos.



Gerenciamento de Projetos

3) A organização pode querer desenvolver as **habilidades de seus** funcionários, por isso pode ser designada uma equipe inexperiente para um projeto, a fim de que essa equipe **aprenda e ganhe** experiência.



Gerenciamento de Projetos

Planejamento de Projeto: O gerenciamento eficaz de um projeto de software depende de um planejamento apurado do andamento do projeto. O gerente de projeto deve prever os **problemas** que podem surgir e preparar soluções experimentais para esses problemas. Um plano traçado no início do projeto deve ser utilizado como **guia** para esse projeto.



Gerenciamento de Projetos

- O plano inicial deve **ser o melhor possível**, em face das informações disponíveis. Ele deve evoluir à medida que o projeto seja desenvolvido e melhores informações se tornem disponíveis.



Gerenciamento de Projetos: Tipos de Planos

Plano	Descrição
Plano de Qualidade	Descreve os procedimentos para teste de qualidade que serão em utilizados em um projeto.
Plano de Validação	Descreve a abordagem, os recursos e o método utilizado para a validação do sistema.
Plano de Gerenciamento de Configuração	Descreve os procedimentos de gerenciamento e as estruturas a serem utilizadas.
Plano de Manutenção	Prevê os requisitos de manutenção do sistema, os custos de manutenção e o esforço necessário.
Plano de Desenvolvimento em Equipe	Descreve como as habilidades e a experiência dos membros da equipe de projeto serão desenvolvidas.



Gerenciamento de Projetos

O plano de projeto: Define os recursos disponíveis para o projeto, a estrutura analítica do trabalho e uma programação para realizar o trabalho.

- Em algumas organizações, o plano de projeto é um documento único que incorpora todos os diferentes **tipos de plano**. Em outros casos, o plano de projeto se ocupa exclusivamente do **processo de desenvolvimento**.



Gerenciamento de Projetos

- Os detalhes do plano de projeto variam, dependendo do tipo de projeto e da empresa. Contudo, a maioria dos planos **deve incluir as seguintes seções:**
 - 1) ***Introdução:*** Descreve com brevidade os objetivos do projeto e define as restrições (por exemplo, orçamento e prazo) que afetam o gerenciamento do projeto.



Gerenciamento de Projetos

- 2) *Organização de projeto*: Descreve o modo como a equipe de desenvolvimento é organizada, as pessoas envolvidas e seus papéis na equipe.
- 3) *Análise de riscos*: Descreve possíveis riscos de projeto, a probabilidade de surgir esses riscos e as estratégias propostas para a redução deles.



Gerenciamento de Projetos

4) *Requisitos necessários de hardware e software*: Descreve o hardware e o software de apoio exigidos para realizar o desenvolvimento. Se o hardware tiver de ser comprado, deverão ser incluídos os prazos de entrega e as estimativas de preço.

5) *Estrutura analítica*: Descreve a divisão do trabalho em atividades e identifica os marcos e os produtos a serem entregues com cada atividade.



Gerenciamento de Projetos

6) *Programação de projeto*: Descreve as dependências entre atividades, o tempo estimado requerido para atingir cada marco e a alocação de pessoas nas atividades.

7) *Mecanismos de monitoramento e de elaboração de relatórios*: Descreve os relatórios de gerenciamento que devem ser produzidos, quando eles devem ser produzidos e quais mecanismos de monitoramento são utilizados.



Gerenciamento de Projetos

“O plano de projeto deve ser **revisado** regularmente durante o projeto. Algumas partes, como a programação do projeto, serão freqüentemente modificadas; outras partes serão mais estáveis”.



Gerenciamento de Projetos

Marcos e produtos a serem entregues: Os gerentes precisam de informações. Como o **software é intangível**, essas informações somente podem ser fornecidas como documentos que descrevem o estado do software que está sendo desenvolvido.

- Sem essas informações, é impossível julgar o **progresso e as estimativas de custos** e as programações não podem ser atualizadas.



Gerenciamento de Projetos

- Quando se planeja um projeto, uma série de *marcos* deve ser estabelecido; um marco é o ponto final de uma atividade de processo de software. A cada marco deve haver uma saída formal, como *um relatório*.
- Os *relatórios de marcos* não precisam ser documentos extensos, eles podem ser um *breve relatório* das atividades de projeto realizadas.



Gerenciamento de Projetos

Programação de Projeto: É uma tarefa necessária para os engenheiros de projeto. Os gerentes estimam o tempo e os recursos exigidos para completar as atividades e os organizam em uma seqüência coerente.

- A menos que o projeto em programação seja similar a um projeto anterior, **estimativas precedentes** são uma base incerta para uma nova programação de projeto.



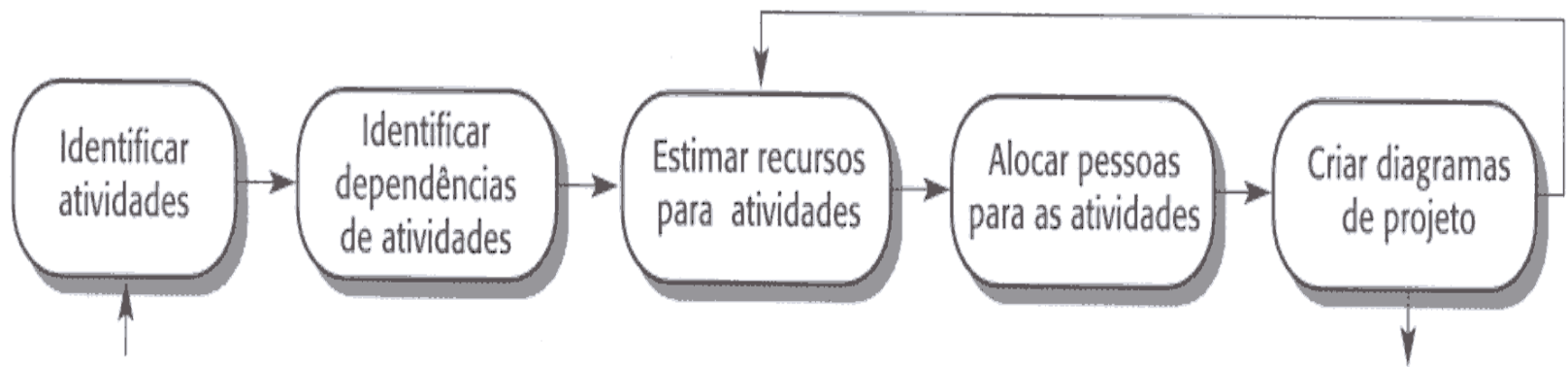
Gerenciamento de Projetos

- A estimativa de programação é ainda mais complicada pelo fato de que diferentes projetos podem utilizar **diferentes métodos** de projeto e linguagens de implementação.



Gerenciamento de Projetos

- A programação de projeto envolve dividir o trabalho total de um projeto em **atividades distintas** e avaliar o tempo necessário para completar essas atividades



Requisitos de software

Diagramas de atividades e diagramas de barras



Gerenciamento de Projetos

- **Diagramas de barras e redes de atividades:** São notações gráficas utilizadas para ilustrar a programação de projeto.
 - Os diagramas de barras mostram **quem é responsável** por cada atividade e para quando está programado o início e o término da atividade.
 - As redes de atividades mostram a **dependência entre as diferentes atividades** que constituem o projeto.



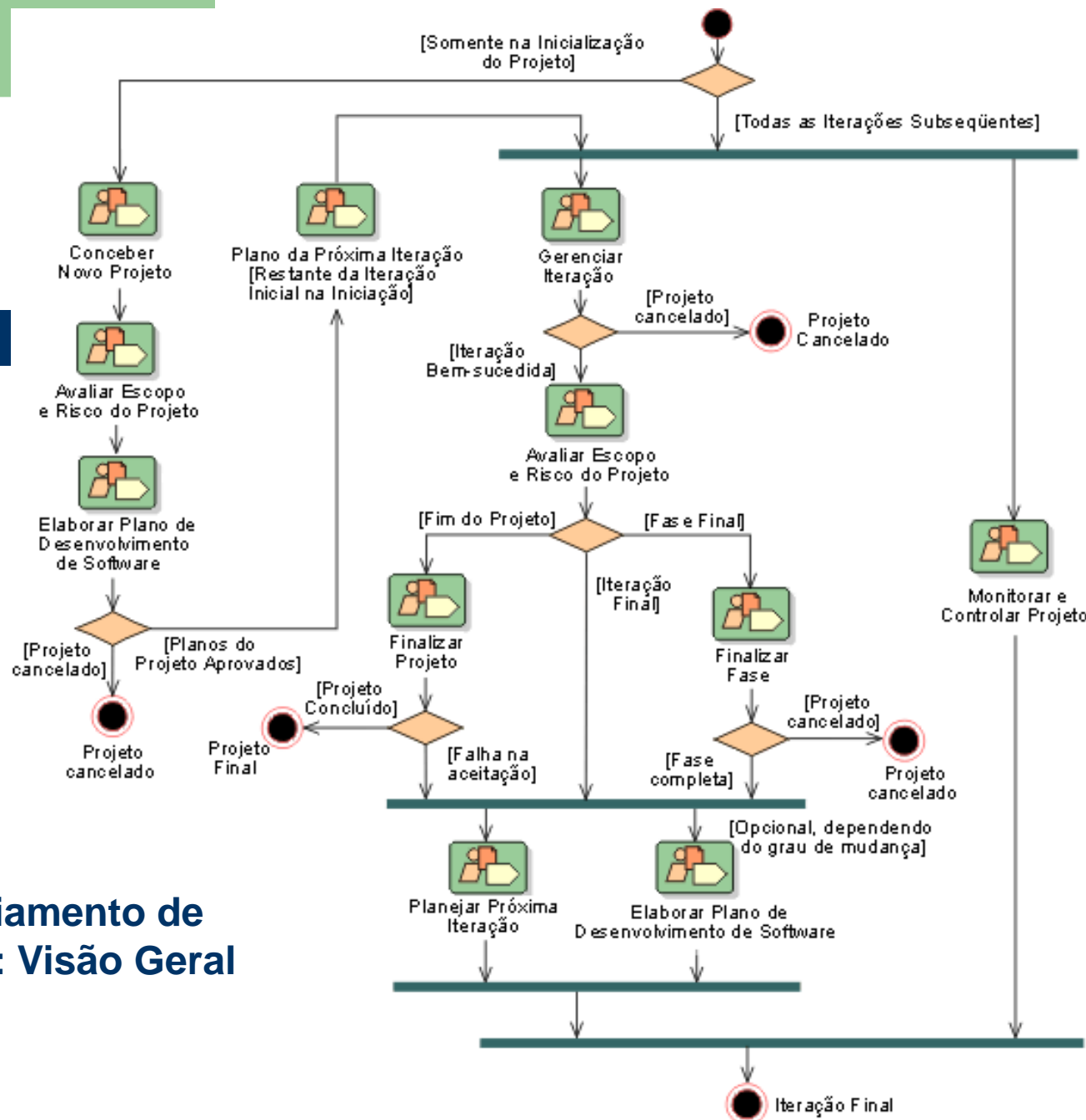
Gerenciamento de Projetos

A finalidade do Gerenciamento de Projeto é:

- Fornecer um **framework** para gerenciar projetos intensivos de software.
- Fornecer diretrizes práticas para planejar, montar a equipe, executar e monitorar os projetos.
- Fornecer um **framework** de gerenciamento de risco.



Gerenciamento de Projeto: Visão Geral



Gerenciamento de Projetos:

Gerenciamento de riscos

Gerenciamento de riscos: Uma importante tarefa de um **gerente de projeto** é prever os **riscos** que podem afetar a programação do projeto ou a qualidade do software em desenvolvimento e tomar as medidas necessárias para evitar esses riscos.

- Os resultados da análise de riscos devem ser documentados **no plano de projeto**, juntamente com uma análise das conseqüências da ocorrência de algum fator de risco.



Gerenciamento de Projetos:

Gerenciamento de riscos

- Identificar riscos e traçar planos para minimizar seus efeitos sobre o projeto é o que se chama de **gerenciamento de riscos**.
- As categorias de riscos podem ser definidas como:
 - 1) **Riscos relacionados ao projeto** :são os riscos que afetam a programação ou os recursos do projeto.



Gerenciamento de Projetos :

Gerenciamento de riscos

2) **Riscos relacionados ao produto** :são os riscos que afetam a qualidade ou o desempenho do software que está em desenvolvimento.

3) **Riscos para os negócios**: são os riscos que afetam a organização que está desenvolvendo ou adquirindo o software.



Gerenciamento de Projetos :

Gerenciamento de riscos

- Um **risco** é uma variável que, em sua distribuição normal, pode ter um valor que comprometa ou elimine o sucesso de um projeto. Em termos simples, um **risco** é tudo que possa se encontrar no **caminho do sucesso**, e que atualmente é desconhecido ou incerto.
- **Sucesso** é atender ao conjunto completo de todos os requisitos e limitações mantidos como expectativas do projeto.



Gerenciamento de Projetos :

Gerenciamento de riscos

- O gerenciamento de riscos é importante para projetos de software, devido às incertezas inerentes que a maioria dos projetos enfrentam.
 - Os riscos podem surgir como decorrência de **requisitos mal** definidos, de dificuldades em **estimar o prazo e os recursos** necessários, da dependência de habilidades individuais e de **mudanças nos requisitos**, em razão de modificações nas necessidades do cliente



Gerenciamento de Projetos :

Gerenciamento de riscos

- O processo de gerenciamento de riscos envolve vários estágios:
 - 1) *Identificação de riscos*: São identificados os possíveis riscos de projeto, produto e negócios.
 - 2) *Análise de riscos*: São avaliadas as possibilidades e as consequências da ocorrência desses riscos.



Gerenciamento de Projetos :

Gerenciamento de riscos

3) *Planejamento de riscos*: São traçados planos para enfrentar os riscos, seja evitando-os seja minimizando seus efeitos sobre o projeto.

4) *Monitoramento de riscos*: O risco é constantemente avaliado e os planos para a diminuição de riscos são revisados, à medida que mais informações sobre eles se tornam disponíveis.



Gerenciamento de Projetos :

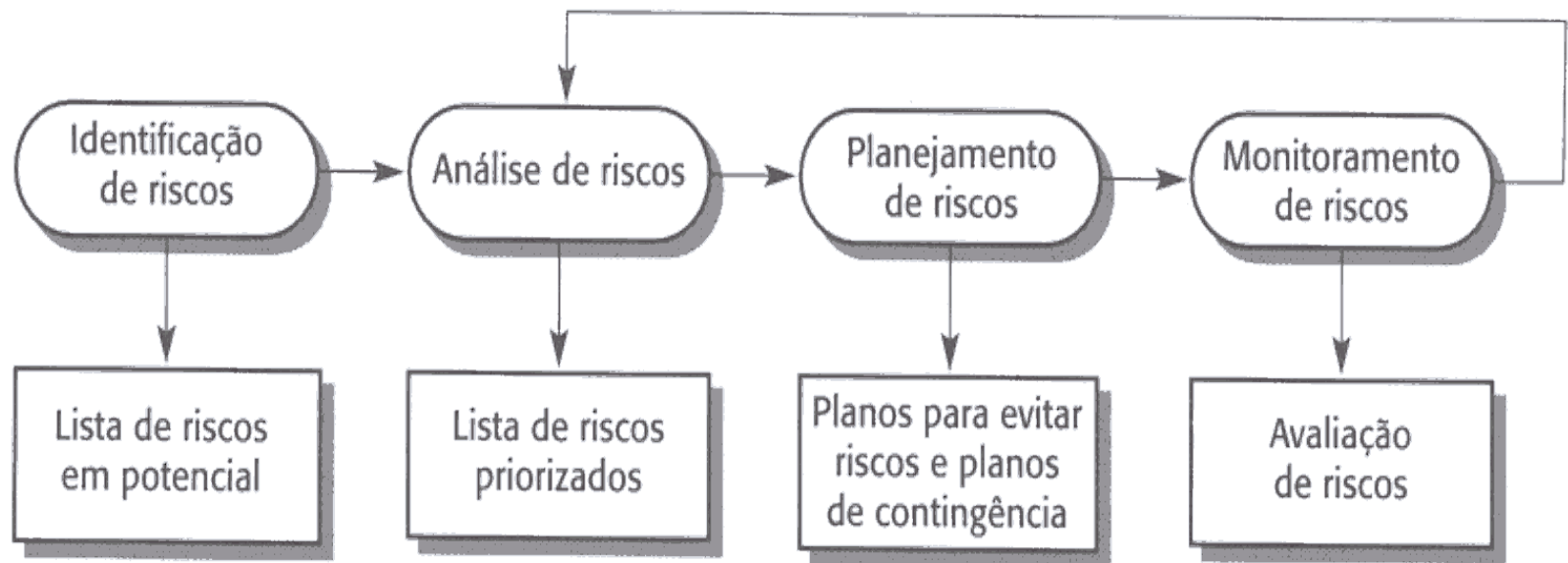
Gerenciamento de riscos

- O processo de gerenciamento de riscos, como todos os outros planejamentos de projeto, é um **processo iterativo**, que continua ao longo do projeto.
- Os resultados do processo de gerenciamento de riscos **devem ser documentados** em um plano de gerenciamento de riscos. Essa fase deve incluir uma discussão **sobre os riscos** apresentados pelo projeto, uma **análise desses riscos** e **os planos** que são necessários para gerenciá-los.



Gerenciamento de Projetos : Gerenciamento de riscos

- O processo de gerenciamento de riscos.



Gerenciamento de Projetos :

Gerenciamento de riscos

- **Identificação de riscos:** É o primeiro estágio do **gerenciamento de riscos**; se preocupa em descobrir os possíveis riscos existentes no projeto.
 - Em princípio, esses riscos não devem ser avaliados ou priorizados. Os riscos com conseqüências muito pequenas ou com muito pouca probabilidade não são normalmente **levados em consideração**.



Gerenciamento de Projetos :

Gerenciamento de riscos

- A identificação de riscos pode ser realizada como um processo em equipe, utilizando-se **brainstorming**, ou pode simplesmente ter como base a **experiência de um gerente**.
- Usar uma lista de possíveis tipos de riscos:
 - 1) **Riscos quanto à tecnologia** :São os riscos que se originam de tecnologias de software ou de hardware, que são utilizadas como parte do sistema em desenvolvimento.



Gerenciamento de Projetos :

Gerenciamento de riscos

- 2) *Riscos quanto ao pessoal*: São os riscos associados às pessoas da equipe de desenvolvimento.
- 3) *Riscos organizacionais* : São os riscos que derivam do ambiente organizacional em que o software está sendo desenvolvido.
- 4) *Riscos quanto às ferramentas* :São os riscos que derivam de **ferramentas CASE** e de outros tipos de software de apoio, utilizados para desenvolver o sistema.



Gerenciamento de Projetos :

Gerenciamento de riscos

- 5) *Riscos quanto aos requisitos* : São os riscos que derivam de modificações nos requisitos do cliente e do processo de gerenciamento da modificação de requisitos.
- 6) *Riscos quanto à estimativa* : São os riscos que derivam das estimativas feitas pelo gerenciamento sobre as características do sistema e os recursos necessários para construí-lo.



Gerenciamento de Projetos :

Gerenciamento de riscos

Análise de riscos : Durante o processo de análise de riscos, cada **risco identificado** é considerado individualmente e é feito um julgamento sobre a probabilidade e a seriedade desse risco.

- Não existe uma maneira fácil de fazer, isso tudo depende do **julgamento e da experiência do gerente** de projeto.



Gerenciamento de Projetos :

Gerenciamento de riscos

- Não é preciso uma avaliação numérica exata, mas a análise deve ter como base uma análise com utilização de **intervalos**:

- 1) A probabilidade do risco pode ser determinada como **muito baixa** (menor do que 10 por cento), **baixa** (10-25 por cento), **moderada** (25-50 por cento), **alta** (50-75 por cento) ou **muito alta** (maior do que 75 por cento).

- 2) Os efeitos do risco podem ser determinados como **catastróficos, sérios, toleráveis ou insignificantes**.



Gerenciamento de Projetos :

Gerenciamento de riscos

Risco	Probabilidade	Efeitos
Problemas financeiros organizacionais forçam reduções no orçamento do projeto.	Baixa	Catastróficos
É impossível recrutar pessoal com as habilidades requeridas para o projeto.	Alta	Catastróficos
Pessoas-chave estão doentes em períodos cruciais do projeto.	Moderada	Sérios
Componentes de software que deviam ser reutilizados contêm defeitos que limitam sua funcionalidade.	Moderada	Sérios



Gerenciamento de Projetos :

Gerenciamento de riscos

Risco	Probabilidade	Efeitos
São propostas mudanças nos requisitos, que exigem significativo trabalho.	Moderada	Sérios
A organização está estruturada de maneira que diferentes gerências são responsáveis pelo projeto.	Alta	Sérios
O banco de dados utilizado no sistema não pode processar tantas transformações por segundo, como esperado.	Moderada	Sérios



Gerenciamento de Projetos :

Gerenciamento de riscos

Risco	Probabilidade	Efeitos
O tempo requerido para desenvolver o software é subestimado.	Alta	Sérios
As ferramentas CASE não podem ser integradas.	Alta	Toleráveis
Os clientes não compreendem o impacto das mudanças nos requisitos.	Moderada	Toleráveis
O treinamento necessário para o pessoal não está disponível.	Moderada	Toleráveis



Gerenciamento de Projetos :

Gerenciamento de riscos

Risco	Probabilidade	Efeitos
A taxa de solução de efeitos é subestimada.	Moderada	Toleráveis
O tamanho do software é subestimado.	Alta	Toleráveis
O código gerado pelas ferramentas CASE é ineficiente.	Moderada	Insignificante



Gerenciamento de Projetos :

Gerenciamento de riscos

Planejamento de riscos: O processo de planejamento de riscos considera cada um dos **riscos mais importantes** que foram identificados e define estratégias para gerenciá-lo.

- Mais uma vez, não há nenhum processo simples que possa ser seguido para estabelecer **planos de gerenciamento** de riscos. Isso depende do julgamento e da **experiência do gerente de projeto**.



Gerenciamento de Projetos :

Gerenciamento de riscos

- As estratégias classificam-se em três categorias:
 - 1) **Estratégias preventivas:** Seguir essas estratégias significa que a probabilidade de o risco surgir será reduzida.
 - 2) **Estratégias de minimização:** Seguir essas estratégias significa que o impacto do risco será reduzido.
 - 3) **Planos de contingência:** Seguir essas estratégias significa que, **se o pior acontecer**, você está preparado e tem uma estratégia pronta para lidar com o caso.



Risco	Estratégia
Problemas financeiros organizacionais	Prepare um documento informativo para a alta gerência, mostrando como o projeto presta uma contribuição muito importante para os objetivos da empresa.
Problemas de recrutamento	Alerte o cliente sobre as dificuldades em potencial e a possibilidade de atrasos; investigue a compra de componentes.
Doença de pessoas da equipe	Reorganize a equipe de maneira que haja mais sobreposição de trabalho e, portanto, as pessoas compreendam as tarefas umas das outras.
Componentes defeituosos	Substitua componentes potencialmente defeituosos por componentes comprados e que tenham confiabilidade reconhecida.
Alterações nos requisitos	Extraia informações que podem ser rastreadas, para avaliar o impacto das mudanças nos requisitos, maximize a inclusão de informações no projeto.
Reestruturação organizacional	Prepare um documento informativo para a alta gerência, mostrando como o projeto presta uma contribuição muito importante para os objetivos da empresa.
Desempenho do banco de dados	Investigue a possibilidade de comprar um banco de dados com maior desempenho.
Prazo de desenvolvimento subestimado	Investigue a compra de componentes e verifique o uso de um gerador de programas.



Gerenciamento de Projetos :

Gerenciamento de riscos

Monitoramento de riscos: envolve avaliar regularmente cada um dos riscos individuais, a fim de decidir se esse risco está se tornando **mais ou menos provável** e se seus efeitos decorrentes se modificaram.



Gerenciamento de Projetos :

Gerenciamento de riscos

Fatores de Risco:

Tipo de risco	Indicadores em potencial
Tecnologia	Atraso na entrega de hardware ou software de apoio, muitos problemas de tecnologia são relatados.
Pessoal	Pessoal pouco motivado, relacionamento insatisfatório entre os membros da equipe, disponibilidade de trabalho.
Organizacional	Fofocas na empresa, falta de iniciativa por parte da alta gerência.
Ferramentas	Relutância de membros da equipe em utilizar ferramentas, reclamações sobre ferramentas CASE, solicitações de estações de trabalho com maior capacidade.
Requisitos	Muitos pedidos de modificações nos requisitos, reclamações do cliente.
Estimativa	Falha no cumprimento do programa estabelecido, falha em eliminar defeitos registrados.



Gerenciamento de Projetos :

Gerenciamento de riscos

PONTOS-CHAVE

- Um bom gerenciamento de projeto de software é essencial para que os projetos sejam desenvolvidos dentro do **prazo e do orçamento**.
- O gerenciamento de software é diferente dos outros gerenciamentos de engenharia. O **software é intangível**. Os projetos podem ser novos ou inovadores, de modo que não haja **nenhuma experiência** prévia para orientar seu gerenciamento.



Gerenciamento de Projetos :

Gerenciamento de riscos

PONTOS-CHAVE

- Os gerentes de software têm diversos papéis: planejamento de projeto, estimativa e programação. O planejamento e a estimativa são **processos iterativos**, que continuam ao longo do projeto.
- Um marco de projeto é o resultado previsto de uma atividade em que algum **relatório de progresso** deve ser apresentado à gerência. Os marcos devem ocorrer regularmente ao longo de um projeto.



Gerenciamento de Projetos :

Gerenciamento de riscos

PONTOS-CHAVE

- A programação de projeto envolve a criação de várias representações gráficas de parte do plano de projeto:
 - **diagramas de atividades**(mostram o inter-relacionamento de atividades de projeto), e **diagramas de barras**(mostram a duração das atividades).



Gerenciamento de Projetos :

Gerenciamento de riscos

PONTOS-CHAVE

- Os principais riscos de projeto devem ser **identificados e avaliados**, a fim de estabelecer sua probabilidade e suas conseqüências para o projeto.
 - Os riscos devem ser explicitamente **discutidos** em cada reunião sobre o progresso do projeto.



MBA em Engenharia de Software a Distância

**Disciplina:
Análise e Gerência de Requisitos**

Gerenciamento de Requisitos - O quê e como Gerenciar

- O **gerenciamento de requisitos** é um modelo sistemático para encontrar, **documentar, organizar e rastrear** os requisitos variáveis de um sistema.
- **Gerência de Requisitos** estabelece as diretrizes para um **comum entendimento** entre o cliente e os **requisitos de seu projeto** de software, que serão conduzidos pela equipe do desenvolvimento.



Gerenciamento de Requisitos - O quê e como Gerenciar

- Um **requisito** é definido como: Uma **condição ou uma capacidade** com a qual o sistema deve estar de acordo.
- A definição formal de gerenciamento de requisitos trata-se de um **modelo sistemático** para:
 - Identificar, organizar e documentar os requisitos do sistema, e;
 - Estabelecer e manter acordo entre **o cliente e a equipe do projeto** nos requisitos do sistema.



Gerenciamento de Requisitos - O quê e como Gerenciar

- Os principais itens para o **gerenciamento** eficiente de requisitos incluem manter:
 - uma declaração clara dos **requisitos**, juntamente com atributos aplicáveis para cada tipo de requisito e **rastreabilidade**.
- A **rastreabilidade** é a capacidade de **rastrear** um **elemento do projeto** a outros **elementos correlatos**, especialmente aqueles relacionados a requisitos.

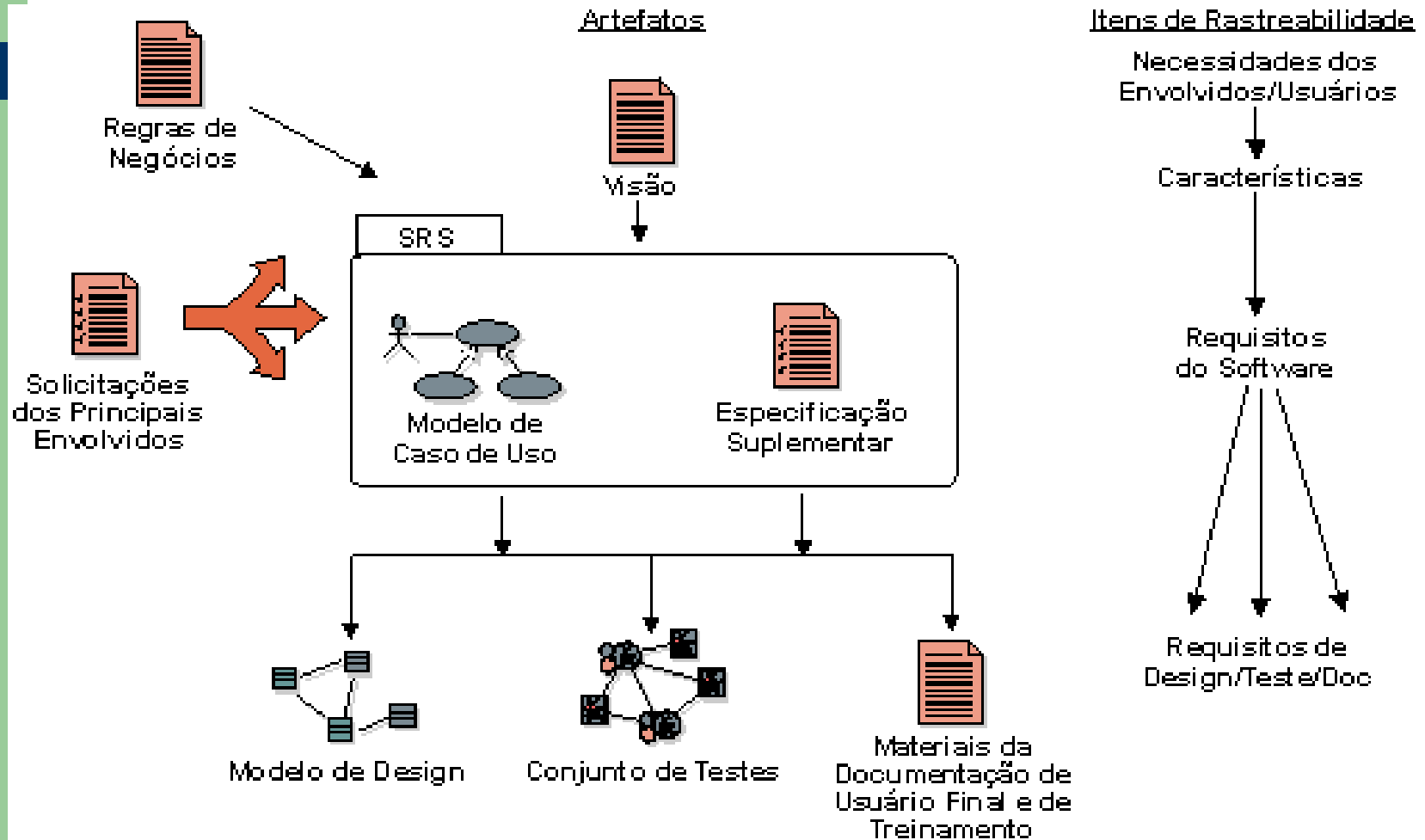


Gerenciamento de Requisitos - O quê e como Gerenciar

- Os elementos do projeto envolvidos na rastreabilidade são chamados de **itens de rastreabilidade**.
- Os itens **típicos de rastreabilidade** incluem diferentes tipos de requisitos, elementos de modelo de design e de análise, artefatos de **testes** (conjuntos de testes, casos de teste, etc.) e **material de treinamento e documentação de suporte a usuário final**.



Gerenciamento de Requisitos - O quê e como Gerenciar



Gerenciamento de Requisitos - O quê e como Gerenciar

A finalidade da **rastreabilidade** é ajudar a :

- 1) compreender a origem dos requisitos;
- 2) gerenciar o escopo do projeto;
- 3) gerenciar mudanças nos requisitos;
- 4) avaliar o impacto no projeto da mudança em um requisito;
- 5) avaliar o impacto da falha de um teste nos requisitos;



Gerenciamento de Requisitos - O quê e como Gerenciar

6) verificar se todos os requisitos do sistema são desempenhados pela implementação;

7) verificar se o aplicativo faz apenas o que era esperado que ele fizesse.

- A **rastreabilidade** ajuda a **compreender e gerenciar** como as informações fornecidas sobre **os requisitos** são convertidas em um **conjunto de necessidades-chave dos envolvidos/usuários** e características do sistema.



Gerenciamento de Requisitos - O quê e como Gerenciar

- A coleta de requisitos pode parecer uma tarefa bem precisa. Nos projetos reais, contudo, você encontrará dificuldades porque:
 - Nem sempre os requisitos são óbvios e podem vir de várias fontes.
 - Nem sempre é fácil expressar os requisitos claramente em palavras.
 - Existem diversos tipos de requisitos em diferentes níveis de detalhe.



Gerenciamento de Requisitos - O quê e como Gerenciar

- O número de requisitos poderá impossibilitar a gerência se não for controlado.
- Os requisitos **estão relacionados** uns com os outros, e também com o **produto liberado** do processo de engenharia do software.
- Os requisitos **têm propriedades exclusivas** ou valores de propriedade. Por exemplo, eles não são igualmente importantes nem igualmente fáceis de cumprir.



Gerenciamento de Requisitos - O quê e como Gerenciar

- Há várias partes interessadas, o que significa que os **requisitos precisam ser gerenciados** por grupos de pessoas de diferentes funções.
- Os requisitos são alterados.
- Então, que habilidades você precisa desenvolver em sua organização para ajudá-lo a gerenciar essas dificuldades? É importante dominar as **seguintes habilidades**:



Gerenciamento de Requisitos - O quê e como Gerenciar

- **Análise do Problema:** É feita para compreender os problemas e as **necessidades iniciais** dos envolvidos, e propor soluções de alto nível.
 - Durante a análise do problema, são **reconhecidos os problemas reais e quais são os envolvidos**. Definir quais são, as fronteiras da solução e as restrições de negócios da solução.



Gerenciamento de Requisitos - O quê e como Gerenciar

- **Noções Básicas sobre as Necessidades dos Envolvidos:** Os requisitos vêm de várias fontes, como clientes, parceiros, usuários finais e peritos do domínio.
 - É preciso saber o melhor modo de determinar quais devem ser as fontes, como obter acesso a essas fontes e qual a melhor forma de levantar as informações delas.



Gerenciamento de Requisitos - O quê e como Gerenciar

- **Definição do Sistema:** Definir o sistema significa **traduzir e organizar** as necessidades dos envolvidos em descrições significativas do sistema a ser construído. No início da definição do sistema, ocorre o seguinte:
 - as decisões sobre o que **constitui um requisito**;
 - o formato de documentação;
 - a **formalidade da linguagem**, o grau de especificidade dos requisitos (quantos e com que detalhe);



Gerenciamento de Requisitos - O quê e como Gerenciar

- a prioridade das solicitações e o esforço estimado;
- os riscos **técnicos e de gerenciamento**;
- o escopo inicial. Parte dessa atividade pode incluir **modelos de design e protótipos** iniciais diretamente relacionados aos mais importantes requisitos dos envolvidos.



Gerenciamento de Requisitos - O quê e como Gerenciar

- **Gerenciamento do Escopo de um Projeto:**
Para gerenciar com eficiência um projeto, é necessário **priorizar os requisitos**, com base no **retorno dado por todos os envolvidos**, e gerenciar o seu escopo.
 - Vários projetos têm seus **desenvolvedores trabalhando nas características** que eles acham interessantes e desafiadoras, em vez de estarem **concentrados** desde o início em tarefas que aliviam **algum risco** no projeto ou estabilizam a arquitetura do aplicativo.



Gerenciamento de Requisitos - O quê e como Gerenciar

- **Refinamento da Definição do Sistema:** A definição detalhada do sistema precisa ser apresentada de maneira que os envolvidos possam **entendê-la, concordar** com ela e **sair** dela.
 - Precisa abordar não apenas a **funcionalidade**, mas também a **compatibilidade com os requisitos**, a usabilidade, a confiabilidade, o desempenho, a capacidade de suporte e de manutenção.



Gerenciamento de Requisitos - O quê e como Gerenciar

- **Gerenciamento dos Requisitos Variáveis:** Não importa o quão cuidadoso você seja sobre a **definição dos seus requisitos**, sempre haverá mudanças.
 - O que torna complexo o gerenciamento dos requisitos variáveis não é apenas que um **requisito mudado** implicará mais ou menos tempo gasto na implementação de uma determinada característica nova, mas também que a **mudança em um requisito terá impacto em outros requisitos**.

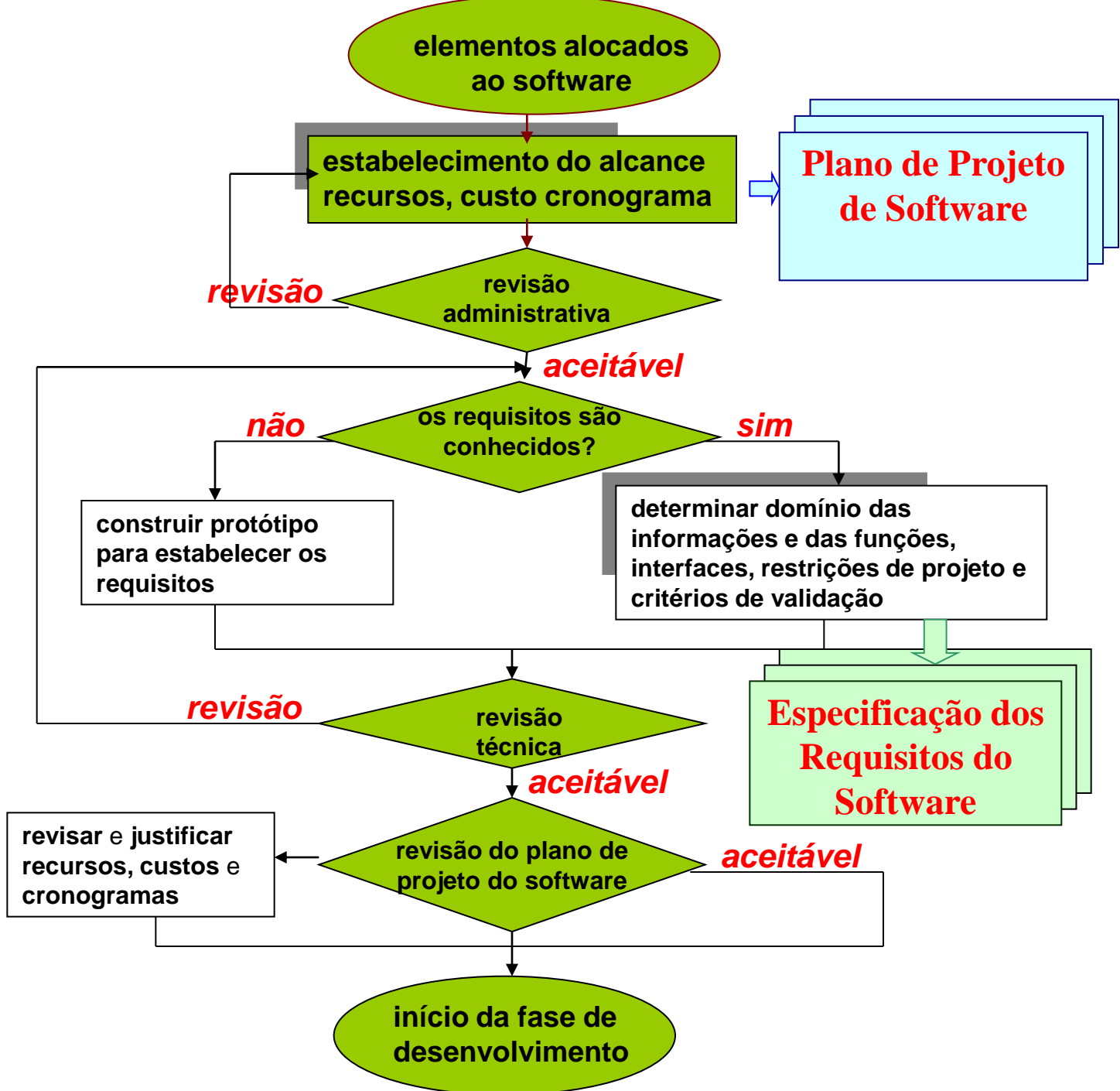


Gerenciamento de Requisitos - O quê e como Gerenciar

- O gerenciamento de mudança inclui:
 - estabelecer uma linha de base;
 - determinar quais dependências são importantes de serem rastreadas;
 - estabelecer a rastreabilidade entre itens correlatos e o controle de mudança.



Fase de Análise de Requisitos



MBA em Engenharia de Software a Distância

**Disciplina:
Análise e Gerência de Requisitos**



Gerenciamento de Requisitos baseado em Use Cases

Documentação associada ao modelo de casos de uso

- O modelo de casos de uso captura os requisitos funcionais e força o desenvolvedor a pensar em como os **agentes externos interagem** com o sistema.
 - Esse modelo corresponde somente aos **requisitos funcionais**. Outros tipos de requisitos (desempenho, interface, segurança, regras do negócio etc.) que fazem parte do documento de requisitos de um sistema **não são considerados** pelo modelo de casos de uso.



Gerenciamento de Requisitos baseado em Use Cases

- Itens da especificação de requisitos que podem estar relacionados com o modelo de casos de uso:
 - Regras de negócio;
 - Requisitos de interface;
 - Requisitos de desempenho.



Gerenciamento de Requisitos baseado em Use Cases

Regras de negócio

- Regras de negócio são políticas, condições ou restrições que devem ser consideradas na execução dos processos existentes em uma organização.
 - As regras de negócio constituem uma parte importante dos processos organizacionais porque descrevem a maneira como a organização funciona.
 - O termo "regra de negócio" é utilizado mesmo em organizações não caracterizadas como empresariais.



Gerenciamento de Requisitos baseado em Use Cases

- Cada organização pode ter várias regras do negócio. As regras do negócio de uma organização são normalmente identificadas nas fases de levantamento de **requisitos de análise**.
 - Essas regras **são documentadas** no chamado ***modelo de regras do negócio***. Nesse modelo, as regras são categorizadas. Cada regra normalmente recebe um identificador (assim como acontece para cada caso de uso).



Gerenciamento de Requisitos baseado em Use Cases

- A descrição do modelo de regras do negócio pode ser feita utilizando-se texto informal, ou alguma forma de estruturação. Alguns exemplos **de regras do negócio**:
 - O valor total de um pedido é igual à soma dos totais dos itens do pedido acrescido de 10% de taxa de entrega.
 - Um professor só pode estar lecionando disciplinas para as quais esteja habilitado.
 - Um cliente do banco não pode retirar mais de R\$ 1.000 por dia de sua conta.



Gerenciamento de Requisitos baseado em Use Cases

Documentação de uma regra de negócio

Nome	Quantidade de inscrições possíveis (RN01)
Descrição	Um aluno não pode se inscrever em mais de seis disciplinas por semestre letivo.
Fonte	Coordenador de escola de informática.
Histórico	Data de identificação: 12/07/2002.



Gerenciamento de Requisitos baseado em Use Cases

Requisitos de desempenho:

- O modelo de casos de uso também não considera *requisitos de desempenho*. Um requisito de desempenho define características relacionadas à operação do sistema.
 - Exemplos: número esperado de transações por unidade de tempo, tempo máximo esperado para uma operação, volume de dados que deve ser tratado, etc.



Gerenciamento de Requisitos baseado em Use Cases

Conectando casos de uso a requisitos de desempenho

Identificador do caso de uso	Frequência da utilização	Tempo máximo esperado
CSU01	5/mês	Interativo
CSU02	15/dia	1 segundo
CSU03	60/dia	Interativo
CSU04	180/dia	3 segundos
CSU05	600/mês	10 segundos
CSU07	500/dia durante 10 dias seguidos	10 segundos



Gerenciamento de Requisitos baseado em Use Cases

Requisitos de interface

- A especificação dos requisitos de um sistema pode também **conter uma seção que descreva** os requisitos de interface do sistema.
 - Por exemplo, o cliente pode ter definido restrições específicas com respeito à interface do sistema: cor, estilo, interatividade etc.
- Os requisitos de interface podem estar relacionados a um ou mais casos de uso do sistema.



Gerenciamento de Requisitos baseado em Use Cases

Modelo de casos de uso no processo de desenvolvimento:

Caso de Uso (Use Case)

- Captura a funcionalidade do ponto de vista do usuário;
- Desenvolvido nos primeiros estágios do desenvolvimento;
- Modelam os **requisitos funcionais** do sistema.
- Propósito:
 - Especificar o contexto do sistema;
 - Capturar os requisitos do sistema;

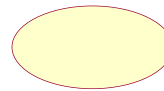


Gerenciamento de Requisitos baseado em Use Cases

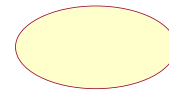
- Um **caso de uso** é uma **seqüência de ações** realizadas por um sistema, que gera um resultado observável de valor para um determinado ator.
- Um caso de uso representa **quem** faz o **que** (**interage**) com o sistema, sem considerar o comportamento interno do sistema.
- Notação UML:



Manter Aula



Emitir Relatório



Manter Disciplina



Manter Material Didático



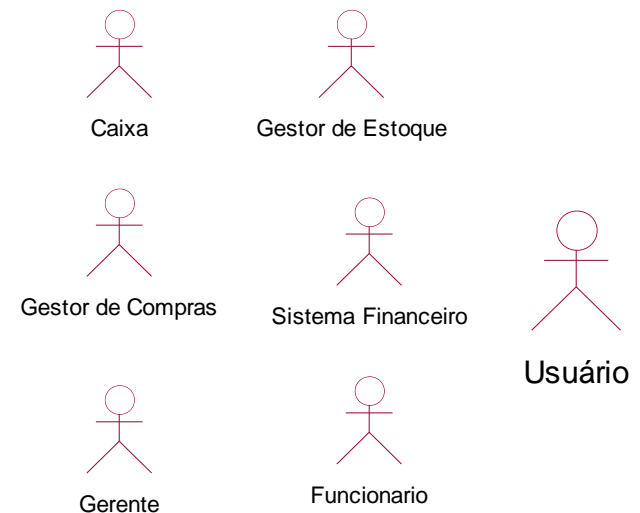
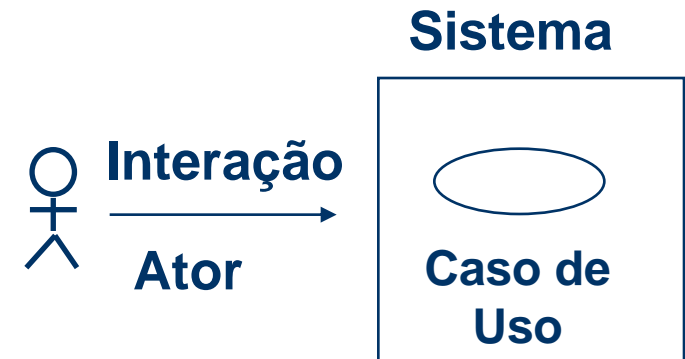
Acessar Material Didático



Gerenciamento de Requisitos baseado em Use Cases

Atores

- Elemento externo que interage com o sistema;
 - “Externo”: atores *não* fazem parte do sistema.
 - “interage”: um ator troca informações com o sistema.
- Representam um papel;
- Iniciam casos de uso.



Gerenciamento de Requisitos baseado em Use Cases

Categorias de atores:

- **Pessoas** (Empregado, Cliente, Gerente, Almoxtarife, Vendedor, etc);
- **Organizações** (Empresa Fornecedora, Agência de Impostos, Administradora de Cartões, etc);
- **Outros sistemas** (Sistema de Cobrança, Sistema de Estoque de Produtos, etc).
- **Equipamentos** (Leitora de Código de Barras, Sensor, etc.).

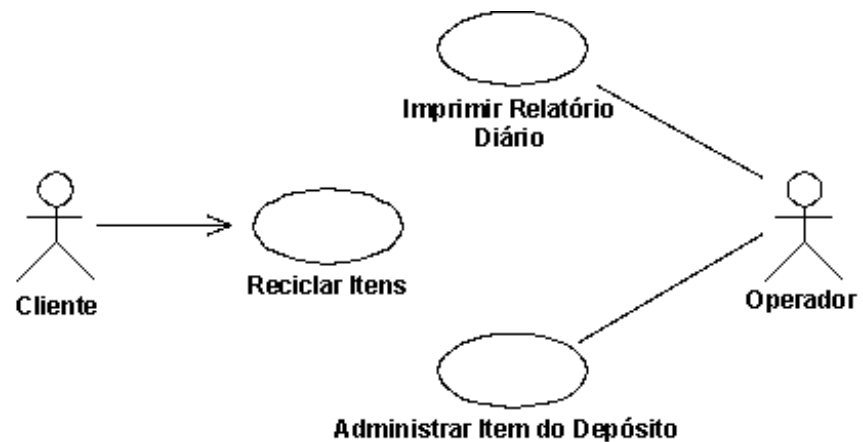


Gerenciamento de Requisitos baseado em Use Cases

Modelo de Casos de Uso

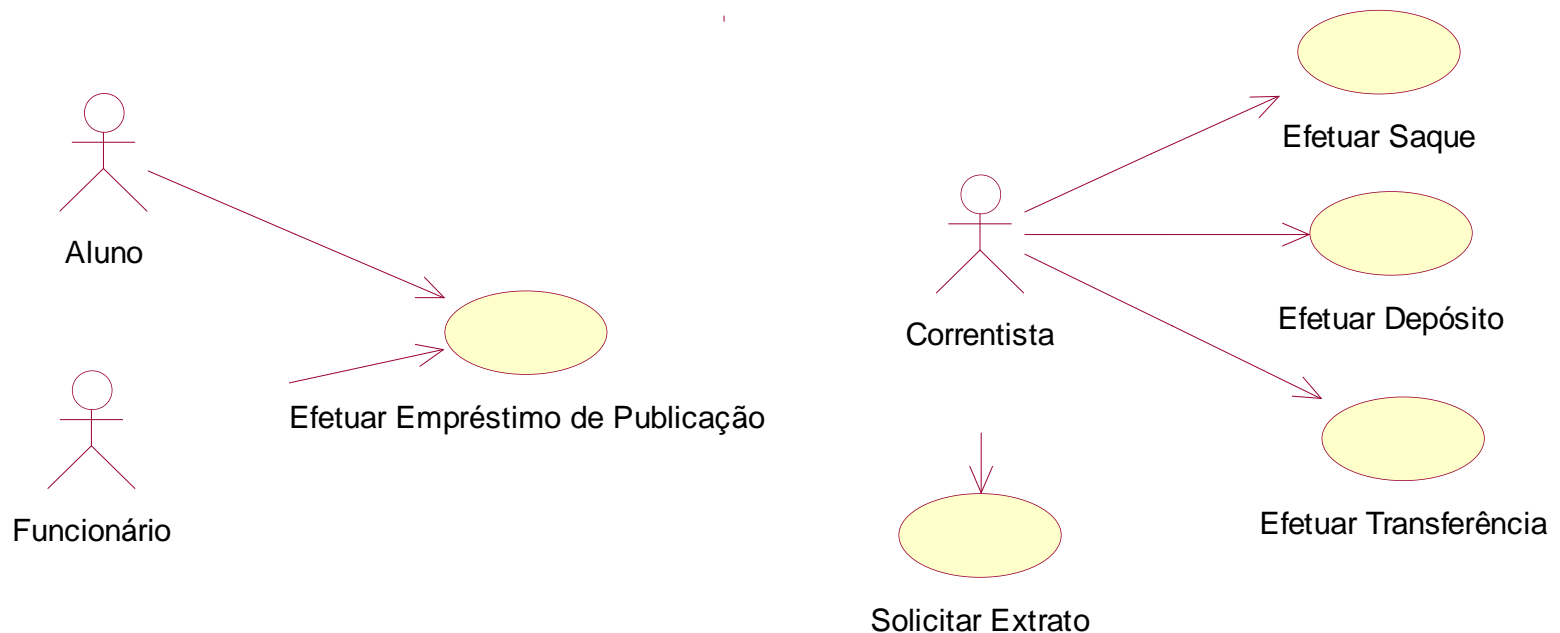
- O **modelo de casos de uso** é um modelo que descreve os requisitos de um sistema em termos de casos de uso.

modelo de casos de uso para o Sistema da Máquina de Reciclagem



Gerenciamento de Requisitos baseado em Use Cases

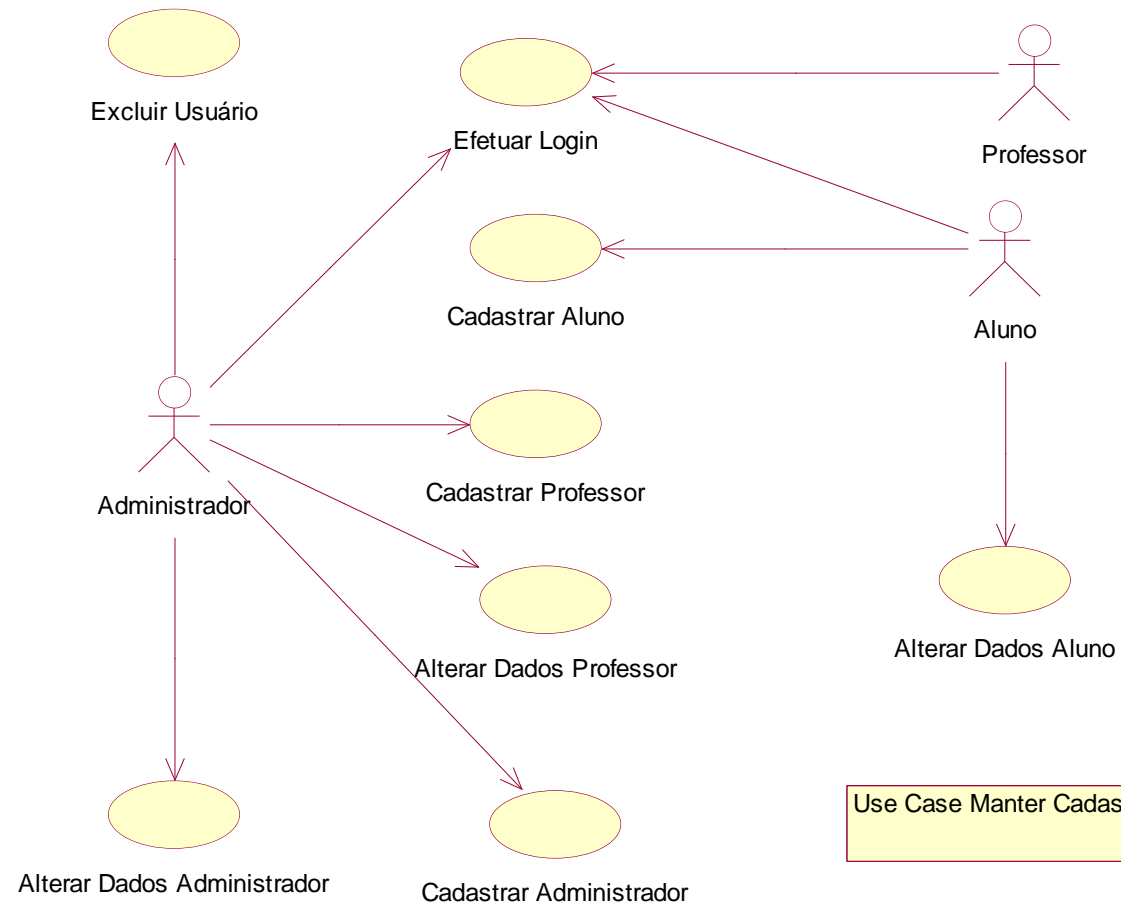
Diagramas de **casos de uso** são criados para **visualizar** os **relacionamentos entre atores e casos de uso**.



Gerenciamento de Requisitos baseado em Use Cases

Modelo de Casos de Uso

- É uma técnica usada para **descrever e definir os requisitos funcionais** de um sistema.



Use Case:	Cadastrar aluno
Sumário:	Este caso de uso inicia-se quando o aluno não possui cadastro e tem a necessidade de cadastrar-se no CEAED.
Atores:	Aluno
Pré-condições:	Não se aplica a este caso.
Descrição:	<p>O aluno seleciona no sistema a opção cadastro;</p> <ol style="list-style-type: none"> 1.O sistema disponibiliza ao aluno o formulário de cadastro de aluno, onde os campos com asteriscos são considerados de caráter obrigatório: login, *senha, *nome, *sexo, RG, CPF *endereço (descrição do endereço, descrição do complemento, número, *bairro, *CEP), *cidade, *estado, *UF, *telefone residencial, telefone celular, e-mail, *escolaridade, *tipo de necessidade especial, *nome da mãe, nome do pai, *RG do responsável, CPF do responsável, Nome do responsável e *data de nascimento; 2.O aluno terá acesso aos botões: “Gravar” e “Fechar”; 3.Caso o aluno escolha à opção “Fechar”; 4.O sistema abortará a gravação dos dados e retornará à tela inicial do CEAED; 5.Caso o aluno escolha à opção “Gravar”; 6.O sistema validará os dados cadastrados emitindo a mensagem “<i>Os dados foram cadastrados com sucesso!</i>”, e retornará a tela inicial do aluno, aplicando saudação ao referente Aluno.
Alternativas:	<ol style="list-style-type: none"> 1.No item dois, caso o aluno não preencha os campos considerados obrigatórios, o sistema emitirá uma mensagem de notificação, “<i>Preencher campos obrigatórios</i>”, e retornará ao item dois da descrição de caso de uso; 2.Caso o aluno tenha a necessidade de alterar algum dado antes da inserção na base de dados, este o fará apontando o cursor do mouse em cima do campo em questão e clicando sobre o mesmo, o sistema disponibilizará o cursor intermitente no campo referenciado. 3.A opção “Fechar”, será realizada no ícone “X”. 4.Caso o Aluno tenha escolhido um subitem ou um botão por engano, ele terá a opção de fechar, através do botão “Fechar”, e o sistema sairá da opção escolhida, e retornará para a tela anterior.

