



Tecnologia em Análise e Desenvolvimento de Sistemas

Engenharia de Software II

Engenharia de Requisitos

Prof. Claudemir Santos Pinto
claudemir.santos2@fatec.sp.gov.br

Engenharia de Requisitos

Definições

- É o **processo** de estabelecer quais são os serviços requeridos pelos interessados de um sistema e as restrições para seu uso e desenvolvimento.
- Requisitos são obtidos através de uma atividade de comunicação, posteriormente analisados, formalmente documentados, e subsequentemente implementados.

- A Engenharia de Requisitos (ER) é uma sub-área da Engenharia de Software, que estuda o processo de produção e gerência dos requisitos que o software deverá atender.
- Trabalha junto aos clientes durante a fase de elicitação ou levantamento dos requisitos e perpassa todas as fases do processo de desenvolvimento do software.

- Fornece métodos, técnicas e ferramentas que forneçam suporte adequado às tarefas de produção e gerência dos requisitos do sistema.
- A Engenharia de Requisitos foi estabelecida como disciplina independente em 1993, quando da criação do International Symposium on Requirements Engineering (RE'93)

- A Engenharia de Requisitos engloba os processos de **produção e gerência de requisitos.**
- Produção => Gerência de Requisitos

Introdução

Importância

- Requisitos constituem a base para a definição da arquitetura do sistema, para a implementação propriamente dita, para geração dos casos de testes e para validação do sistema junto ao cliente.
- Os custos da identificação e correção de erros na fase de levantamento de requisitos é bem menor que nas fases seguintes.

Tanto o desenvolvedor quanto o cliente assumem um papel ativo na engenharia de requisitos de software.

- Cliente: tenta formular uma descrição, às vezes nebulosa do software de que precisa, incluindo dados, funções e comportamentos.
- Desenvolvedor: age como inquisidor, consultor, solucionador de problemas e negociador



Como o cliente explicou...



Como o líder de projeto entendeu...



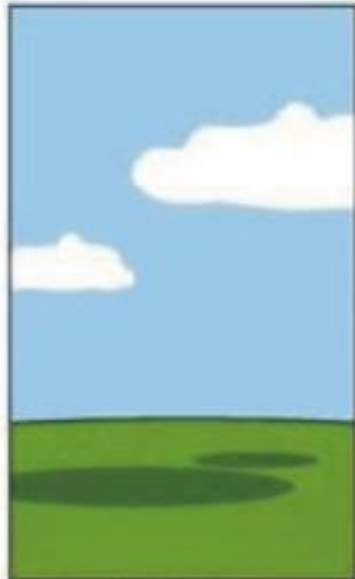
Como o analista projetou...



Como o programador construiu...



Como o consultor de negócios descreveu...



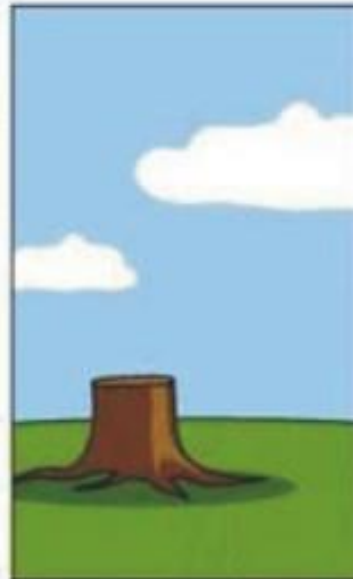
Como o projeto foi documentado...



Que funcionalidades foram instaladas...



Como o cliente foi cobrado...



Como foi mantido...



O que o cliente realmente queria...

Análise e especificação de requisitos pode parecer tarefa simples. Entretanto...

- O conteúdo da comunicação é muito grande
- Grande chance de má interpretação e má informação
- Cliente: “Eu sei que você acredita que entendeu o que pensa que eu disse, mas não estou certo de que você reconhece que o que você ouviu não é o que eu quis dizer”

- Quais são os problemas reais?
 - Muitas vezes o cliente possui apenas uma vaga ideia do que realmente quer / precisa
 - O desenvolvedor fica ansioso para prosseguir com a “vaga idéia” com a intenção de “encontrar os detalhes durante o restante do processo”
 - O cliente modifica continuamente os requisitos
 - O desenvolvedor sofre com as mudanças, as quais levam a erros de especificação e programação
- E o processo continua...

Outras definições para Engenharia de Requisitos:

- “Engenharia de Requisitos pode ser definida como o processo sistemático de desenvolvimento de requisitos através de um processo cooperativo de análise onde os resultados das observações são codificados em uma variedade de formatos e a acurácia (exatidão) das observações é constantemente verificada.” Klaus Pohl (2010)

Outras definições para Engenharia de Requisitos:

- “é o uso sistemático de princípios, técnicas e ferramentas comprovadas para a análise, documentação, evolução continuada das necessidades do usuário e especificação do comportamento externo de um sistema para satisfazer as necessidades do usuário que sejam efetivas em termos de custo. Note que como em todas as disciplinas de engenharia, a engenharia de requisitos não é conduzida de um modo esporádico, aleatório ou sujeito a azares, mas, ao contrário, é o uso sistemático de abordagens comprovadas.” - Donald Reifer (1994)

Por que Engenharia de Requisitos?

B.C. by Johnny Hart



Processo de Produção de Requisitos

- Ian Sommerville, quatro etapas:
 - Elicitação, análise, negociação e validação
- Julio Cesar Sampaio do Prado Leite, **etapas essenciais:**
 - Elicitação, modelagem e análise (validação e verificação)
- Karl Wieggers, adiciona uma quarta etapa:
 - Especificação, que foca na documentação de informação relativa a rastreabilidade dos requisitos.

PROCESSO “ESSENCIAL”:

- Entender o problema: **ELICITAÇÃO**
 - Captura/descobrimento de requisitos, comunicação com clientes, aquisição de referências.
 - Utilizar técnicas para elicitar os requisitos: questionários, entrevistas, documentos...
- Estudar o problema: **ANÁLISE**
 - Verificação e Validação de modo a garantir a acuracidade das observações e informações levantadas durante o processo.
- “Desenhar” a solução: **MODELAGEM**
 - Codificação, utilizando-se algum tipo de representação persistente que garanta o acesso posterior aos requisitos e suas origens.
 - Representar nosso entendimento dos problemas utilizando técnicas para modelagem: Ex: MER, UML

Entender o problema: **Elicitação**

- Identificar as fontes de informação
- Coleta de dados/fatos

Fontes de Informação

- Atores do Universo de Informações
 - Clientes
 - Usuários
 - Desenvolvedores
- Documentos
- Livros
- Sistemas de Software similares / concorrentes



Fontes de Informação

- Quem é o cliente?
 - Quem é o dono do sistema?
 - Existe alguma solução (pacote) disponível?
 - Quais são os livros relacionados à aplicação em discussão?
-
- Existe a possibilidade de reutilizar o software?

Coleta de Dados

- Entrevistas;
- Reuniões;
- Observação;
- Leitura de documentos;
- Aplicação de Questionários;
- Bases de Requisitos não funcionais



Entrevistas

- +
 - contato direto com atores
 - possibilidade de validação imediata

- -
 - diferenças culturais
 - conhecimento tácito

(Conhecimento Tácito)

- É aquele conhecimento que é trivial para o entrevistado e não o é para o entrevistador.
- Por ser trivial nunca é lembrado como importante e, portanto, não é transmitido ao entrevistador, que, não sabendo, não pode perguntar.



Conhecimento Tácito

- Perguntar: “porquê?” Ex:

“A cafeteira deve ser feita de aço”

- qual a razão disto?
- pode me explicar porquê?
- qual o pensamento atrás disto?

Conhecimento Tácito

“Porque se for de vidro pode quebrar”

- Requisito real:

A cafeteira deve ser feita de material inquebrável

- Plástico
- Poliuretano
- Até mesmo aço

Atenção

- Os usuários misturam a solução com os requisitos.
- Separar NECESSIDADE da SOLUÇÃO proposta ou atual!

Exercício

- “a cafeteira tem uma luz vermelha que pisca quando está ligada, quando a água chega na temperatura certa ela fica ligada (sem piscar)”
 - Quais seriam as perguntas do tipo “porque” que poderiam ser utilizadas nesta situação?
 - Quais seriam os “reais” requisitos?
- Dica: Separar “requisito” de “solução/implementação”

Reuniões

- Extensão da entrevista;
- Brainstorm;
- Workshop de Requisitos



Reuniões

- +
 - dispor de múltiplas opiniões
 - criação coletiva

- -
 - dispersão
 - custo

Observação

- +
 - baixo custo
 - pouca complexidade da tarefa
- -
 - dependência do observador
 - superficialidade decorrente da pouca exposição ao universo de informações

Leitura de Documentos

- +
 - facilidade de acesso às fontes de informação
 - volume de informação
- -
 - dispersão das informações
 - volume de trabalho requerido para identificação dos fatos

Questionários

- Qualitativo
- Quantitativo
- O que perguntar
 - exige conhecimento mínimo
 - similar a entrevista estruturada

Questionários

DICAS:

- Elucidar a natureza da pesquisa e sua importância;
- Iniciar com perguntas gerais e depois as mais específicas;
- Perguntas claras, sem dupla interpretação;
- Cuidado com os aspectos visuais do questionário;
- Planejar compilação dos dados.



Questionários

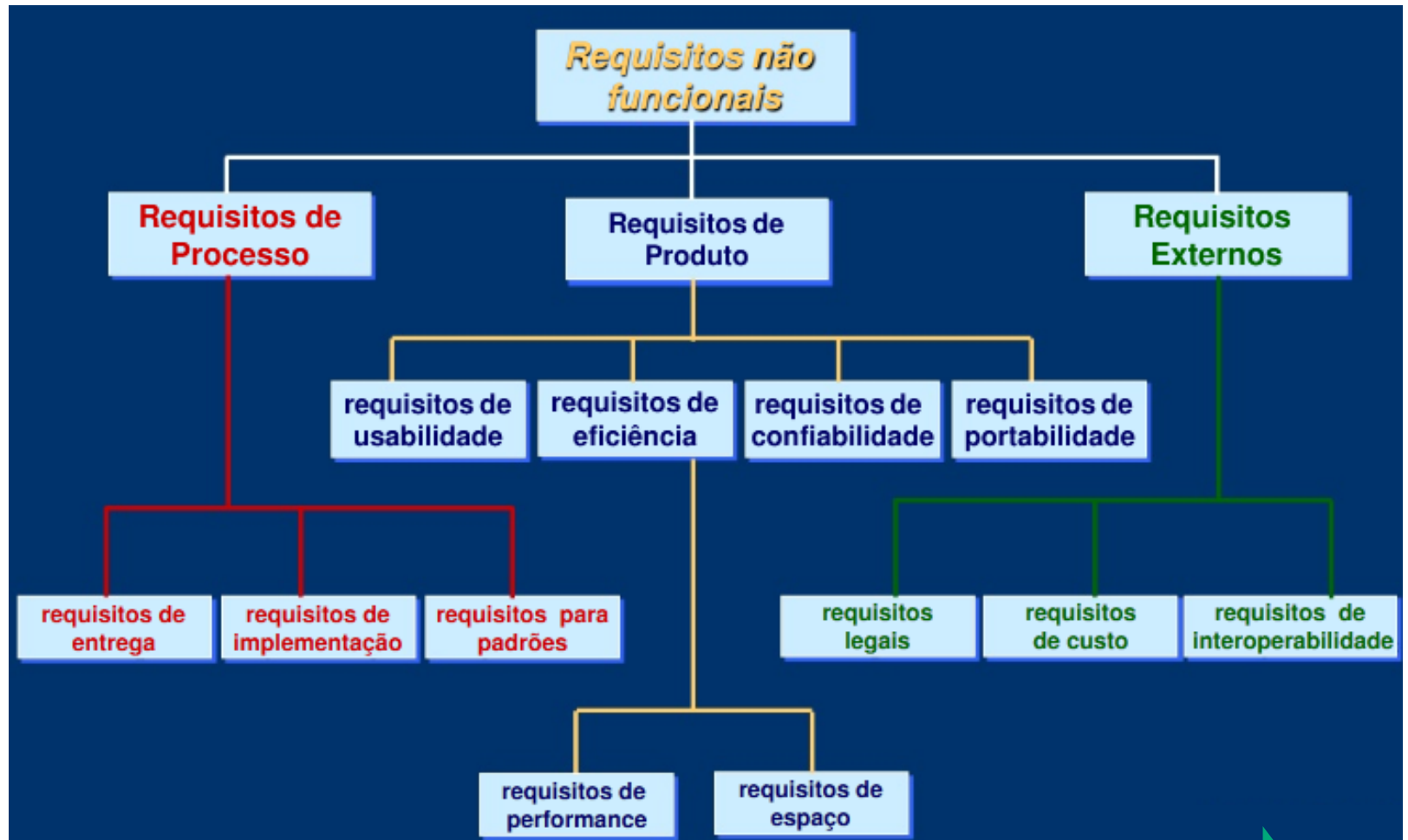
- +

- padronização de perguntas
- tratamento estatístico

- -

- limitação das respostas
- pouca interação/participação

Base de Requisitos não-funcionais



Base de Requisitos não-funcionais

■ +

- reutilização de conhecimento
- antecipação de aspectos implementacionais
- identificação de conflitos

■ -

- falsa impressão de “completeza”

Exercícios de fixação

- Defina: ELICITAÇÃO, MODELAGEM, ANÁLISE
- Dentro do processo de elicitação, quais são as fontes de informação ?
- Quais as formas de coleta de dados ?
- O que é conhecimento tácito? Que cuidados o analista deve ter ao entrevistar um cliente ?
- Como definir o melhor usuário para a entrevista?
- Qual a eficácia das reuniões para o levantamento de requisitos?
- Como formular de forma eficiente o questionário para a entrevista?
- O que é a Base de Requisitos Não-Funcionais ?

Entregar via Teams

Estudar o problema: **ANÁLISE**

Verificação X Validação



Estamos construindo o
produto de maneira certa
(em relação aos
requisitos)



Estamos construindo o
produto certo?
(em relação a necessidade
dos clientes)

Técnicas Informais de verificação

- Leitura ad hoc;
- Inspeções

Leitura ad hoc

- Essa técnica não utiliza nenhuma técnica formal de leitura, cada leitor lê o documento do seu modo, por este motivo ela torna-se dependente da experiência do leitor, e apresenta um grande defeito que é o fato de não existir um procedimento a ser seguido.

Inspeções

- A inspeção visa encontrar erros lendo, entendendo o que o documento descreve e checando através de um checklist as propriedades de qualidade requeridas; é composta por seis fases, que são: Planejamento, Apresentação, Preparação, Reunião de Inspeção, Retrabalho e Acompanhamento

Validação através de Protótipos

- Passivo, ativo ou interativo;
- identifica **atores**, explica **o que** acontece a eles e descreve **como** acontece;
- mais eficazes se o projeto tiver conteúdo inovador ou desconhecido;
- tipo de rascunho, fácil de modificar.

“Desenhar” a solução: **Modelagem**

- Para que servem modelos?
 - Representação
 - Organização
 - Armazenamento
 - Comunicação

Modelagem

- Um modelo é uma simplificação da realidade.
- Modelos são construídos para permitir um melhor entendimento sobre o sistema que está sendo construído.
- Modelos de sistemas complexos são importantes porque não temos capacidade de compreendê-los inteiramente

Tipos de Modelos de Software

Existem diversos tipos de modelos de software. Cada um mais utilizado em uma determinada área do projeto.

Podem ser usados **diagramas** (forma mais usada em Engenharia de Software) ou **protótipos** (usados para apoiar diversos métodos de desenvolvimento)

Princípios de Modelos de Software

1. A escolha do modelo a ser construído influencia diretamente em:
 - Como o problema é atacado
 - Como a solução é delineada
2. Podem ser expressos em diferentes níveis de precisão
3. Os melhores estão conectados à realidade
4. Nenhum modelo único é suficiente: um conjunto pequeno de modelos é mais eficiente ao abordar um tema em questão.

Utilidade de Modelos de Software

1. Visualização do sistema

Permitem uma visão simplificada do produto, permitindo uma maior compreensão do sistema.

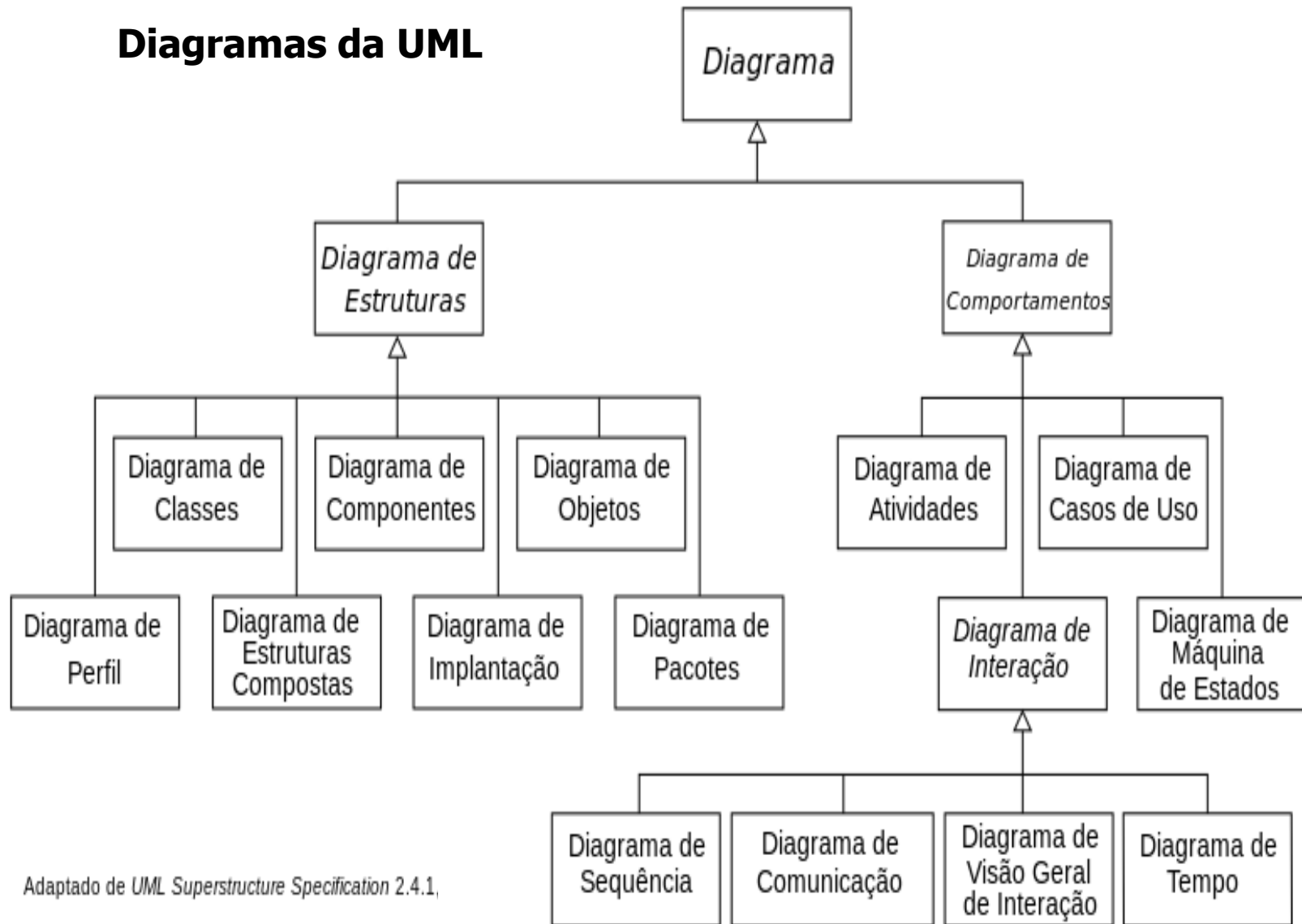
2. Especificação

Permite uma descrição precisa do que será desenvolvido pelos programadores

3. Documentação

Relata o que foi desenvolvido

Diagramas da UML



Adaptado de UML Superstructure Specification 2.4.1,

Protótipos

Página Teste

http://www.ads-es2.com.br

Cadastro

Relatórios

Versão

Data

Código

Nome

Sexo: Escolha ▼
Masculino
Feminino

Salvar Fechar

created with Balsamiq Mockups - www.balsamiq.com

<http://builds.balsamiq.com/b/mockups-web-demo/>

Protótipos

- Vantagens:
 - Barato;
 - amigável, informal e interativo;
 - fornece uma crítica das interfaces do sistema no início do desenvolvimento;
 - fácil de criar e modificar;

Protótipos

- Vantagens:
 - Equívocos entre clientes e desenvolvedores são expostos
 - Serviços esquecidos podem ser detectados e serviços confusos podem ser identificados
 - Pode ser usado para treinamento do usuário e testes do sistema

Protótipos

- Desvantagens:

- O tempo de desenvolvimento de protótipos depende da experiência das pessoas envolvidas. O tempo dos protótipos iniciais pode ser demorado, enquanto se adquire a experiência de como elaborar protótipos de forma rápida e eficiente, o que pode aumentar o custo do projeto final.

Protótipos

- Desvantagens:

- Alguns requisitos, como requisitos de “em tempo real” e requisitos não funcionais podem ser difíceis ou mesmo impossíveis de implementar em um protótipo.

Exercício

- Identificar os atores e respectivas funcionalidades. A partir daí, criar um diagrama de Caso de Uso.
- Em seguida, utilizar uma ferramenta de prototipação online para criar um protótipo de um sistema para vendas de ingressos online.
- Enviar o diagrama de caso de uso e os screenshots do protótipo em PowerPoint.