

Engenharia de Software

Tema da Aula
Análise de Requisitos
Conceitos

Prof. Cristiano R R Portella
portella@widesoft.com.br



Análise de Requisitos

Analisar (v) 1. Decompor um todo em partes, componentes; fazer análise 2. Observar, examinar com minúcia; esquadrinhar 3. Examinar Criticamente.

Requisito (s.m) 1. Condição necessária para obtenção de certo objetivo, ou para preenchimento de certo fim; quesito 2. Exigência necessária para certos efeitos.

Também chamada de Análise de Sistemas ou Engenharia de Requisitos.

Parte dos requisitos obtidos e revisados e do Plano de Projeto. Visa fornecer representações (modelos) de "Informação" e de "função" que:

- Possam ser transformadas (traduzidas) em "projeto" de dados, arquitetura, interface e procedimentos;
- Sirvam como um meio de determinar a qualidade do software.

O Engenheiro de Software constrói modelos usando os requisitos ELICITADOS do usuário.

Inglês (Michaelis)

To Elicit: arrancar (de)

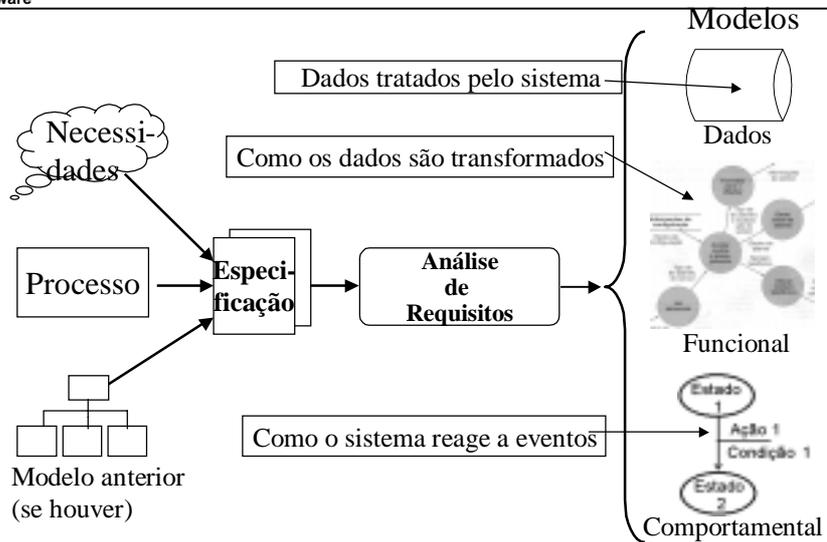
Português (Aurélio)

Eliciar: fazer sair; expulsar.

Para entender e validar os requisitos eliciados, precisamos analisa-los de diferentes pontos de vista:

- Aspectos de dados
- Aspectos funcionais
- Aspectos comportamentais

Análise de Requisitos Fluxo de Atividades



Análise de Requisitos

Objetivos:

1. Descrever o que o usuário requisitou, de maneira técnica;
2. Estabelecer uma base para a criação do Projeto de Software;
3. Definir um conjunto de requisitos que precisam ser validados quando o software estiver pronto.

Princípios de Análise

1. Analisar (dividir) para entender o domínio das Informações de um problema que deve ser representado e compreendido;
2. Criar modelos que descrevam as informações a serem tratadas, as funções que tratarão as informações e o comportamento do sistema que será desenvolvido;
3. Os Modelos (e o problema) devem ser divididos em partições (analisados) de maneira que revele os detalhes em camadas (hierarquicamente);
4. O foco do processo deve mover-se da informação essencial para os detalhes.

Especificação de Requisitos

- ✓ A especificação é o produto (conseqüência) da Análise de Requisitos. Cada requisito exige uma especificação (como deve ser feito e como será validado);
- ✓ É o documento fundamental da Engenharia de Software, pois especifica tudo O QUE deverá ser projetado e construído;
- ✓ Permite que o requisitos seja gerenciáveis (planejamento, controle e feed-back);
- ✓ A revisão dos requisitos é fundamental para garantir que usuário e analista tenham a mesma “percepção” do sistema.

Princípios de Especificação

1. Separar Funcionalidade da Implementação (não conter características de implementação).
2. Usar uma linguagem de Especificação de Sistema Orientada a Processos.
3. A Especificação do deve abranger o sistema do qual o software é um componente.
4. Especificação deve abranger o ambiente(s) no(s) qual(is) o sistema opera

Princípios de Especificação

5. Uma Especificação de Requisitos deve ser um modelo cognitivo.
6. Uma Especificação deve ser operacional
7. A especificação do sistema deve ser tolerante com não-inteireza e ser expansível
8. Uma especificação deve ser localizada e fracamente acoplada

A Engenharia de Software inicia com uma série de tarefas de modelagem, que pretendem obter uma especificação completa de requisitos e projeto compreensivo, que represente o software a ser construído.

A Análise é um conjunto de modelos que será a primeira representação técnica de um sistema. Os modelos predominantes são:

Análise Estruturada (Metodologia Estruturada)

Análise Orientada a Objetos (Met.Orient.Objetos)

}

Modelagem de Dados:

Definir os dados, seus atributos e relacionamentos.

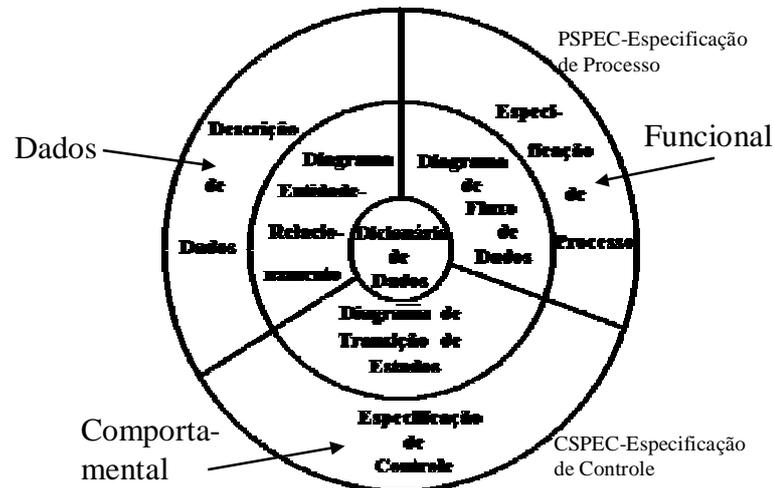
Modelagem Funcional:

Definir como os dados são transformados (processados) dentro do sistema.

Modelagem Comportamental:

Definir como o sistema trata os eventos.

Os elementos do Modelo de Análise



Análise (Estruturada) de Requisitos

Dicionário de Dados:

Repositório que contém a descrição de todos os objetos de dados alimentados ou produzidos pelo software.

Diagrama Entidade-Relacionamento:

Mostra as relações (cardinalidade e modalidade) existentes entre os itens de dados.

Diagrama de Transição de Estados:

Destaca o evento e seu impacto sobre o sistema (fluxo de controle).

Ferramentas CASE (Computer-Aided Software Engineering)

- Aumento de Produtividade e Qualidade

Software de Apoio ao Desenvolvimento de Software:

- Gerenciamento de Projetos (MS-Project)
- Análise e Projeto (Rose; System Architecture; OracleDesigner; ErWin; BpWin; Telelogic, System Architect etc.)
- Geradores de Código (System Architect -> Dr Case)

Software de Apoio ao Desenvolvimento de Software:

- Apoio a Teste, Validação e Simulação
- Gerenciamento de Configuração e Versão
- Documentação de Software

1. Introdução

- a. Referência ao sistema
- b. Objetivos
- c. Restrições ao projeto de software

2. Descrição da informação

- a. Representação do fluxo da informação
- b. Representação do conteúdo da informação
- c. Representação da estrutura da informação
- d. Descrição das interfaces do sistema

3. Descrição funcional

- a. Particionamento funcional
- b. Descrição funcional
 - 1. Texto explicativo do processo
 - 2. Restrições/limitações
 - 3. Requisitos de desempenho
 - 4. Restrições de projeto
 - 5. Diagramas

4. Descrição Comportamental

- a. Estados do Sistema
- b. Eventos e Ações

5. Critérios de validação

- a. Fronteiras de desempenho
- b. Classes de testes
- c. Resposta esperada do software
- d. Considerações especiais

6. Bibliografia

7. Apêndice