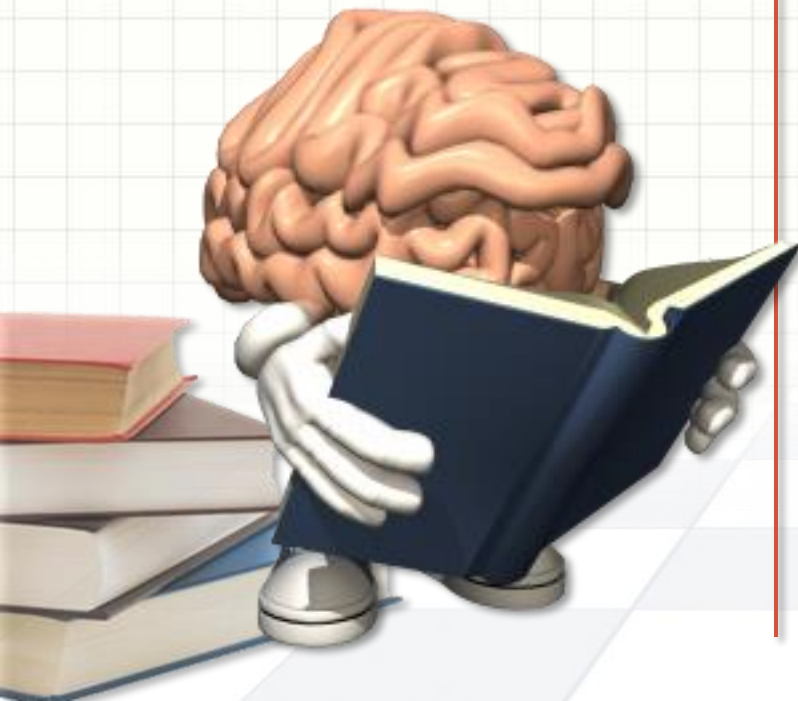


Engenharia de Software I

IES-100

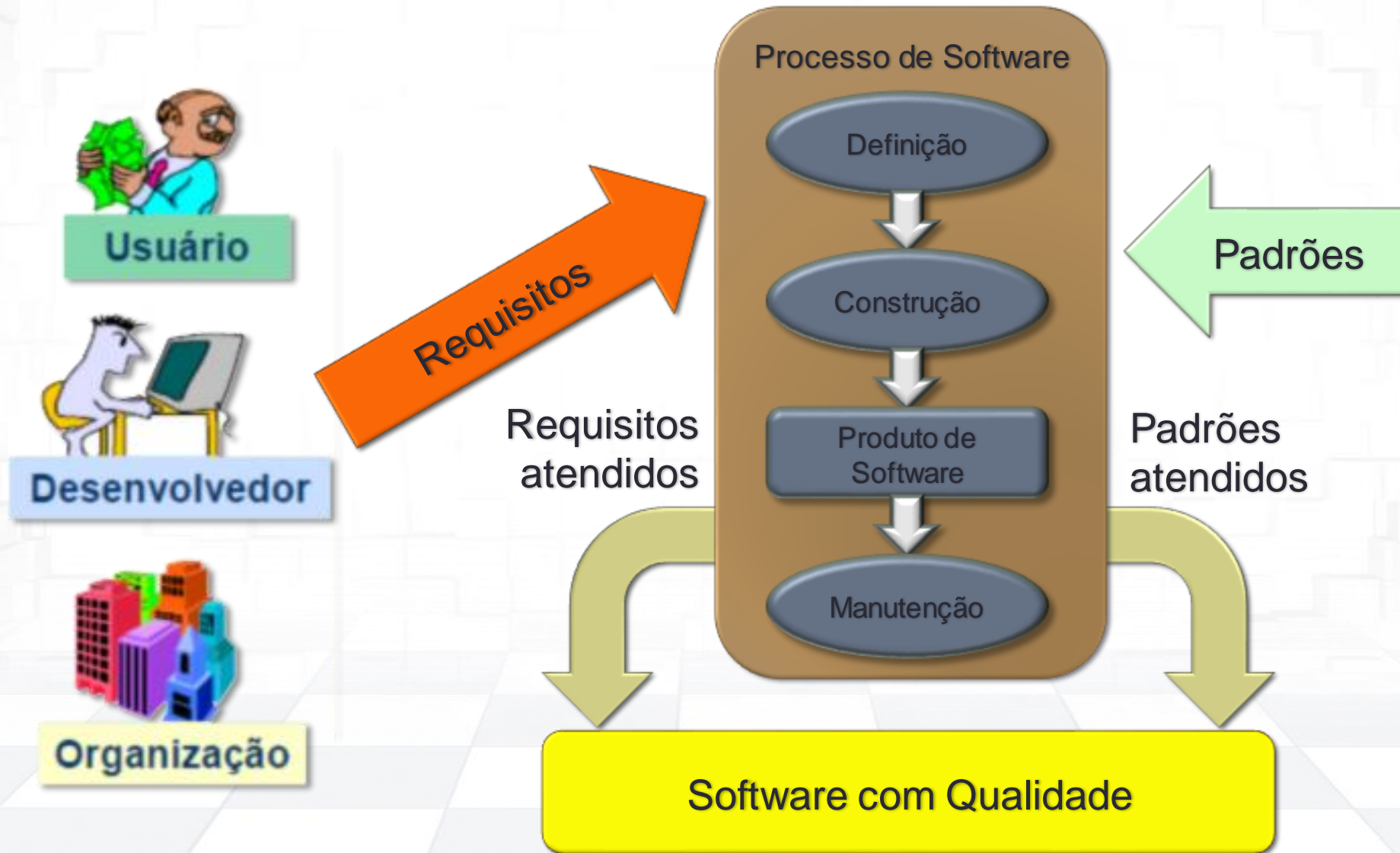
Tópicos

- **Padrão IEEE 830/1998**



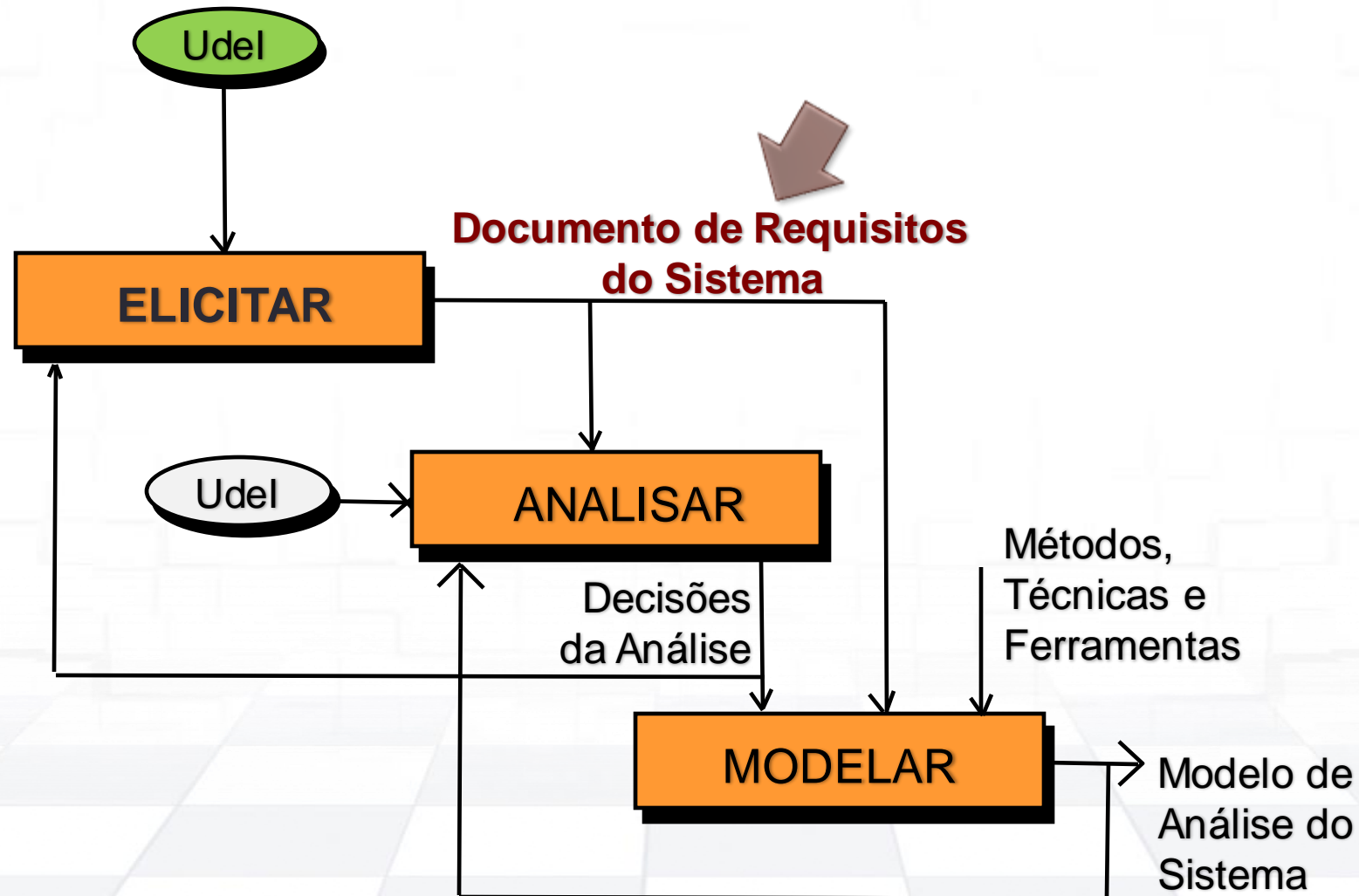
Requisitos de Software

Base da Qualidade



Processo de Definição de Requisitos

Atividades principais



IEEE Std 830/1998

- Práticas recomendadas pela IEEE (Institute of Electrical and Eletronics Engineers) para **especificação de requisitos de software**
- Norma que recomenda abordagens para a Especificação de Requisitos de Software (ERS / SRS – *Software Requirements Specification*)
- ERS
 - Documento que permite ao cliente **descrever** suas necessidades e ao desenvolvedor **compreendê-las**
 - Documento de Requisitos
 - Define todos os **requisitos** que devem compor o software
 - Estabelece uma base para o **acordo** entre clientes e desenvolvedores sobre o que o software fará

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [1/7]

- Quanto à **natureza** da ERS:
 - É uma especificação de produto de software que realiza certas funções em um ambiente específico
- Questões básicas:
 - **Funcionalidade:** O que o software pretende fazer?
 - **Interfaces Externas:** Como o software interage com as pessoas, hardware do sistema, outros hardwares e outros softwares?
 - **Desempenho:** Qual é a velocidade, disponibilidade, o tempo de resposta, o tempo de recuperação das várias funções do software?
 - **Atributos:** Quais são as considerações sobre portabilidade, manutenibilidade, segurança, corretitude etc?
 - **Restrições impostas em uma implementação:** Existe algum padrão requerido, linguagem de programação, políticas de integridade da BD, limitação de recursos, ambientes operacionais, etc?

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [2/7]

- Quanto ao **ambiente** da ERS:
 - A ERS deve estar de acordo com os **requisitos** do sistema
 - Definir **corretamente** os requisitos de software
 - Um requisito de software deve existir devido à natureza da tarefa a ser resolvida ou devido às características especiais do projeto.
 - **Não** descrever quaisquer detalhes da fase de **projeto** ou de **implementação**
 - Descrever apenas **o que**, e não **como**
 - **Não** estabelecer **restrições adicionais**
 - Essas são especificadas de outros documentos, tais como um plano de **garantia de qualidade**.

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [3/7]

- Quanto às **características** da ERS [1/4]:
 - Uma ERS deve:
 - **Ser correta:** Cada requisito **expresso** deve ser **encontrado** também no **software**
 - **Ser não ambígua:** Cada requisito declarado deve possuir **apenas uma interpretação**
 - **Linguagem Natural:** inerentemente ambígua
 - **Linguagens de Especificação de Requisitos** (Linguagem Z, dentre outras): detectam erros sintáticos (estrutura, composição) e de semântica (significado)
 - **Ferramentas de Representação:** que dão suporte aos métodos e linguagens em três perspectivas – objeto, processo e comportamento

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [4/7]

- Quanto às **características** da ERS [2/4]:
 - Uma ERS deve:
 - **Ser completa:** Deve incluir:
 - **Todos** os **requisitos significativos** relacionados à funcionalidade, desempenho, restrições
 - Definição das **respostas** do software para todas as **entradas** e **saídas** de dados
 - **Nomes** completos e **referências** para todas as **tabelas**, **figuras** e **diagramas** e **definição** para todos os **termos** e unidades de medida
 - **Ser consistente:** Não deve haver conflitos entre os requisitos

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [5/7]

- Quanto às **características** da ERS [3/4]:
 - Uma ERS deve:
 - **Possuir grau de importância e/ou estabilidade:** Cada requisito deve identificar seu **grau de importância** ou **estabilidade** em relação a outros (uns são essenciais, outros desejáveis)
 - **Grau de estabilidade:** pode ser expresso em termos do **número** de **mudanças esperadas** para algum requisito
 - baseado na experiência ou conhecimento de eventos que afetam a organização, funções e pessoas apoiadas pelo sistema.
 - **Grau de necessidade:** Essencial, Condicional, Opcional
 - **Ser verificável:** quando for possível checar cada requisito
 - É preciso usar termos concretos e quantidades mensuráveis

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [6/7]

- Quanto às **características** da ERS [4/4]:
 - Uma ERS deve:
 - **Ser modificável:** Requisitos devem ser **facilmente**, **completamente** e **consistentemente** alterados. A ERS deve:
 - Ter organização coerente e fácil de usar com uma tabela de conteúdo e referência cruzada
 - Não ser redundante
 - Expressar cada requisito separadamente
 - **Ser rastreável:** A **origem** de cada um dos requisitos deve ser **clara** e deve facilitar a **referência** de cada requisito na documentação ou desenvolvimento futuro

IEEE Std 830/1998

Considerações para se produzir uma boa ERS [7/7]

- Quanto à **preparação da junção** da ERS:
 - O **cliente** e o **fornecedor** devem trabalhar **juntos** para produzir uma ERS bem escrita e completa
 - Uma ERS pode ser **elaborada durante** o **desenvolvimento** do produto de software.
 - Um **protótipo** pode ser usado como uma maneira de **elicit**ar requisitos de um software

IEEE Std 830/1998

Partes da ERS

- **1. Introdução**
 - 1.1. Objetivo
 - 1.2. Escopo
 - 1.3. Definições, siglas e abreviações
 - 1.4. Referências
 - 1.5. Visão Geral
- **2. Descrição Geral do Produto**
 - 2.1. Perspectiva do produto
 - 2.2. Funções do produto
 - 2.3. Características do Usuário
 - 2.4. Restrições, Suposições e dependências
 - 2.5. Requisitos Adiados
- **3. Requisitos Específicos**
- **4. Apêndices**
- **5. Índice**

Partes da ERS

Capítulo 1: Introdução

- Fornece uma introdução à ERS e uma **descrição** do **software** a ser desenvolvido
- Contém as seguintes seções:
 - 1.1. Objetivo
 - 1.2. Escopo
 - 1.3. Definições, siglas e abreviações
 - 1.4. Referências
 - 1.5. Visão Geral

Partes da ERS

1.1. Objetivo

- Delinear o objetivo da **ERS**.
- Especificar o **público alvo**
 - cliente, analista e desenvolvedor.
- **Obs:**
 - **Não falar do software** neste tópico
 - apenas da ERS (manual do sistema)

Partes da ERS

1.1. Objetivo – Exemplo

O objetivo principal deste manual consiste em documentar os requisitos especificados do software a ser produzido, inteirando o cliente e os desenvolvedores sobre o desenvolvimento e a utilização do software de uma maneira clara e objetiva.

Partes da ERS

1.2. Escopo

- Identificar pelo nome o produto do software a ser produzido (1º parágrafo da seção)
- Explicar **o que o produto de software fará** (requisitos funcionais)
 - e o que não fará (requisitos inversos), se for o caso
- **Descrever a aplicação** do software incluindo **benefícios relevantes**, os **objetivos específicos** e como o software auxilia o processo de negócio
- **Obs:**
 - O escopo deve coincidir com as Funções do Produto;
 - Um escopo bem descrito tem pelo menos 1,5 página de texto, dependendo do tamanho do sistema.

Partes da ERS

1.2. Escopo – Exemplo

O sistema *GymManager* objetiva gerenciar todo o funcionamento de uma academia de ginástica, desde o controle dos funcionários e alunos até o pagamento das mensalidades e respectivos faturamentos.

O software realiza o gerenciamento dos alunos englobando informações pessoais, atividades físicas e situações financeiras, deixando tais informações disponíveis ao gerente e instrutores para devido acompanhamento dos alunos em suas atividades.

Opcionalmente, o acesso dos alunos no estabelecimento pode ser controlado por meio de uma catraca eletrônica, permitindo ou barrando a entrada de acordo com a situação de cada aluno, além de registrar o horário de entrada e de saída.

...

Partes da ERS

1.3. Definições, Siglas e Abreviações

- Fornecer as definições de termos, siglas e abreviações necessárias para interpretar apropriadamente a ERS
 - Podem ser fornecidas por referência a apêndices na ERS ou a outros documentos
- **Exemplo:**
 - Backup: Cópia de segurança dos dados do sistema;
 - WWW: World Wide Web;
 - Calhau: Sobra de espaço em uma página de jornal ou revista que é utilizada para propaganda própria, preenchendo o espaço da página.

Partes da ERS

1.4. Referências

- Fornecer uma lista completa de todos os documentos referenciados
- Identificar cada documento por título, nº, data etc
- Especificar as origens das referências (quem forneceu)
 - Os documentos referenciados devem estar em um Anexo

- **Exemplo:**

Nº	Título	Data de aquisição	Responsável pelo fornecimento
1	Ficha de controle de frequência	27/02/2006	Luiza Meireles (Diretora de serviço)
2	Declaração de encargos de família para imposto de renda	27/02/2006	Luiza Meireles (Diretora de serviço)

Partes da ERS

1.5. Visão Geral

- Descrever o que conterà a ERS
- Explicar como a ERS estará organizada (a partir do capítulo 2)
- Exemplo:

Esta ERS está organizada em capítulos. O Capítulo 2 fornece uma descrição geral do software a ser desenvolvido, contendo uma perspectiva do produto, funções... O Capítulo 3...

Partes da ERS

Capítulo 2: Descrição Geral do Produto

- Descreve fatores gerais do produto e seus requisitos
 - mas não requisitos específicos
- Fornece um background para esses requisitos (que são detalhados na seção 3):
 - 2.1. Perspectiva do Produto
 - 2.2. Funções do Produto
 - 2.3. Características do Usuário
 - 2.4. Restrições, Suposições e Dependências
 - 2.5. Requisitos adiados

Partes da ERS

2.1. Perspectiva do Produto [1/2]

- Deve ser descrita de maneira **resumida**, de forma **textual**, sem detalhamento
 - Aproximadamente **1/2 página**, pois trata-se de uma **descrição geral**.
 - As **interfaces** mencionadas nessa seção serão detalhadas na seção Requisitos de Interface Externa.
 - O produto é colocado em perspectiva com outros produtos relacionados, **podendo** incluir:
 - **Interfaces do Sistema**: com quais outros sistemas o produto de software interage (se houver).
 - **Interfaces do Usuário**: formatos de telas, relatórios ou consulta, formatos de mensagens, acesso por níveis de usuário.
 - **Interfaces de Hardware**: como o produto interage com os dispositivos de hardware; características de configuração

Partes da ERS

2.1. Perspectiva do Produto [2/2]

- **Interfaces de Software:** deve especificar o uso de outros softwares necessários (BD, SO, software p/ capturar imagem etc)
- **Interfaces de Comunicação:** especificar os protocolos de redes locais, protocolos de comunicação para sistemas multicamadas etc
- **Limites de Memória:** especificar as características e os limites de memória primária e secundária (limite mínimo)
- **Operações:** deve especificar requisitos de operações normais e especiais como rotinas de inicialização (definir os níveis de acesso), processamento, backup's e restauração.
- **Requisitos para adaptação de situação:** especificar situações em que o software deve ser adaptado antes da instalação (qualquer sequência de inicialização)

IEEE Std 830/1998

2.2. Funções do Produto [1/3]

- A seção mais importante do capítulo.
- São descritas todas as **funções** (requisitos funcionais) do produto.
- Para cada função, devem ser descritos:
 - Os itens de entrada (dados/informação) necessários;
 - Os itens de saída necessários;
 - As regras de negócio.

IEEE Std 830/1998

2.2. Funções do Produto [2/3]

- Essas funções são classificadas em:
 - **Funções Básicas:** referem-se às **operações CRUD** necessárias para a execução das funções fundamentais. Esse conjunto de operações pode ser denominado **Gerenciar** ou **Manter**.
 - **Funções Fundamentais:** referem-se às **transações de negócio** (movimentações);
 - **Funções de Saída:** referem-se às funções que geram **informações de saída** relevantes para atender às necessidades do cliente (consultas/relatórios com cruzamento de informações). Devem ser descritos os itens de entrada (filtros) e os itens de saída (informação) pertinentes.

IEEE Std 830/1998

2.2. Funções do Produto [3/3]

• Observações:

- Cada função deve ter um **identificador**, a fim de facilitar a **rastreabilidade** desse requisito nesse documento.
 - Sugere-se que seja utilizado RF (requisito funcional) seguido de um *underline*, uma letra indicando se é função básica, fundamental ou saída externa (B, F, S) e um número sequencial.
 - **Ex:** RF_B1 e RF_B2 para funções **básicas**, RF_F1 e RF_F2. para funções **fundamentais** e RF_S1, RF_S2 para funções de **saída** externa.
- **Não** devem ser citados aqui os **campos** das possíveis **tabelas** do sistema, tais como, **códigos** sequenciais criados para facilitar na implementação. Aqui deverão ser citados **apenas** os **itens de informação** relacionados às funções do sistema.
- **Obs:** As funções de **gerenciamento do usuário**, **backup** e **restauração** do sistema **não serão citadas aqui**, uma vez que já foram descritas no item “Perspectiva do Produto”

IEEE Std 830/1998

2.2. Funções do Produto – Exemplo [1/3]

- **Funções Básicas:**

- **RF_B01: Gerenciar Produtos.** Permite que produtos sejam incluídos, pesquisados, alterados e excluídos. Itens de dados: código, nome, descrição, foto, preço, estoque, estoque mínimo, estoque máximo, relação de fornecedores e tipo (perecível/não perecível).
- **RF_B02: Manter Fornecedores. ...**
- **RF_B03: Gerenciar Categorias de Produtos. ...**
- ...

IEEE Std 830/1998

2.2. Funções do Produto – Exemplo [2/3]

- **Funções Fundamentais:**

- **RF_F01: Venda.** O sistema realiza a venda de produtos, sendo que, ao final da mesma, o estoque é automaticamente atualizado. Itens de entrada: código e quantidade. Itens de saída: código, descrição, preço, foto e quantidade, subtotal da venda e total da venda.
- **RF_F02: Pagamento. ...**
- **RF_F03: Pedido de Produtos. ...**
- **RF_F04: Geração de Nota Fiscal Eletrônica. ...**
- **...**

IEEE Std 830/1998

2.2. Funções do Produto – Exemplo [3/3]

- **Funções de Saída:**

- **RF_S01: Relatório Geral de Vendas.** Informa as vendas de um determinado período. Filtro: data inicial e data final. Itens de saída: data da venda, operador de caixa, total da venda e tipo de pagamento.
- **RF_S02: Gráfico de Produtos mais Vendidos. ...**
- **RF_S03: Listagem de Fornecedores mais Cotados. ...**
- ...

IEEE Std 830/1998

2.3. Características do Usuário

- Descrever o **nível educacional** dos **usuários** do sistema, bem como a sua **experiência** e o **conhecimento** sobre **informática** para que seja diagnosticada a necessidade de treinamento específico

- Exemplo:

Os gerentes da empresa do cliente possuem ensino superior e bom conhecimento em informática básica. Já os demais funcionários possuem, no geral, ensino médio e conhecimento razoável em informática básica.

IEEE Std 830/1998

2.4. Restrições, Suposições e Dependências

- Deve fornecer uma descrição geral de qualquer outro item que **limitará** as opções do **desenvolvedor**
 - Ex: Normas reguladoras; Limitações do hardware; Interfaces com outras aplicações; Linguagem de programação; Protocolos; Requisitos de segurança, etc.
- Deve fornecer uma lista de **fatores** que **afetam** os **requisitos** expressos na ERS.
Exemplo:
 - O **limite** para que um certo sistema não tenha sua funcionalidade completa seria a não aquisição do ponto eletrônico.
 - A **suposição** é de que será adquirido o ponto eletrônico.
 - O desempenho total do sistema depende da satisfação dessa suposição, pois a não aquisição do ponto eletrônico fará com que a entrada de dados seja feita manualmente, inserindo somente as exceções do ponto diário, ou seja, a falta dos funcionários.

IEEE Std 830/1998

2.5. Requisitos Adiados

- Identificar, dentre os requisitos **especificados anteriormente**, os que podem ser **adiados** até as **versões futuras** do sistema.
- Exemplo:
 - **Requisitos Adiados:**
 - A função **RF_F04** será implementada na próxima versão do sistema devido a questões de cronograma e tecnologias necessárias.

IEEE Std 830/1998

Capítulo 3: Requisitos Específicos

- Essa seção deve conter **todos** os **requisitos** do software com um nível de **detalhamento suficiente** para possibilitar aos projetistas/desenvolvedores projetar um sistema que atenda a esses requisitos
- Depende do paradigma de desenvolvimento