

**UNIVERSIDADE REGIONAL DE BLUMENAU**

**FERRAMENTA DE GERÊNCIA DE REQUISITOS DE  
SOFTWARE INTEGRADA COM ENTERPRISE ARCHITECT**

Raphael Marcos Batista

Profa. Fabiane Barreto Vavassori Benitti, Dr<sup>a</sup> Eng.

# Roteiro da Apresentação

- Justificativa;
- Objetivos;
- Fundamentação teórica;
- Desenvolvimento do trabalho;
- Conclusão;
- Extensões.

# Justificativa

- Utilizar o padrão IEEE-830-1998 para gerar um documento de ERS.
- Utilizar um ambiente específico que contemple um bom gerenciamento de requisitos.
- Propor a integração entre ferramentas CASE na área de Engenharia de Software.

# Objetivos

- Disponibilizar uma extensão da ferramenta CASE EA para especificação e gerenciamento de requisitos.
- Disponibilizar através da ferramenta uma documentação de requisitos baseada no modelo IEEE-830-1998.

# Fundamentação

- Engenharia de Requisitos.
  - Requisito: descrição abstrata de alto nível de uma funcionalidade ou restrição do sistema.
  - Definição: conjunto de técnicas de levantamento, documentação e análise.
  - Processo da Engenharia Requisitos é concluir com êxito um acordo entre quem solicita e quem desenvolve.

# Fundamentação

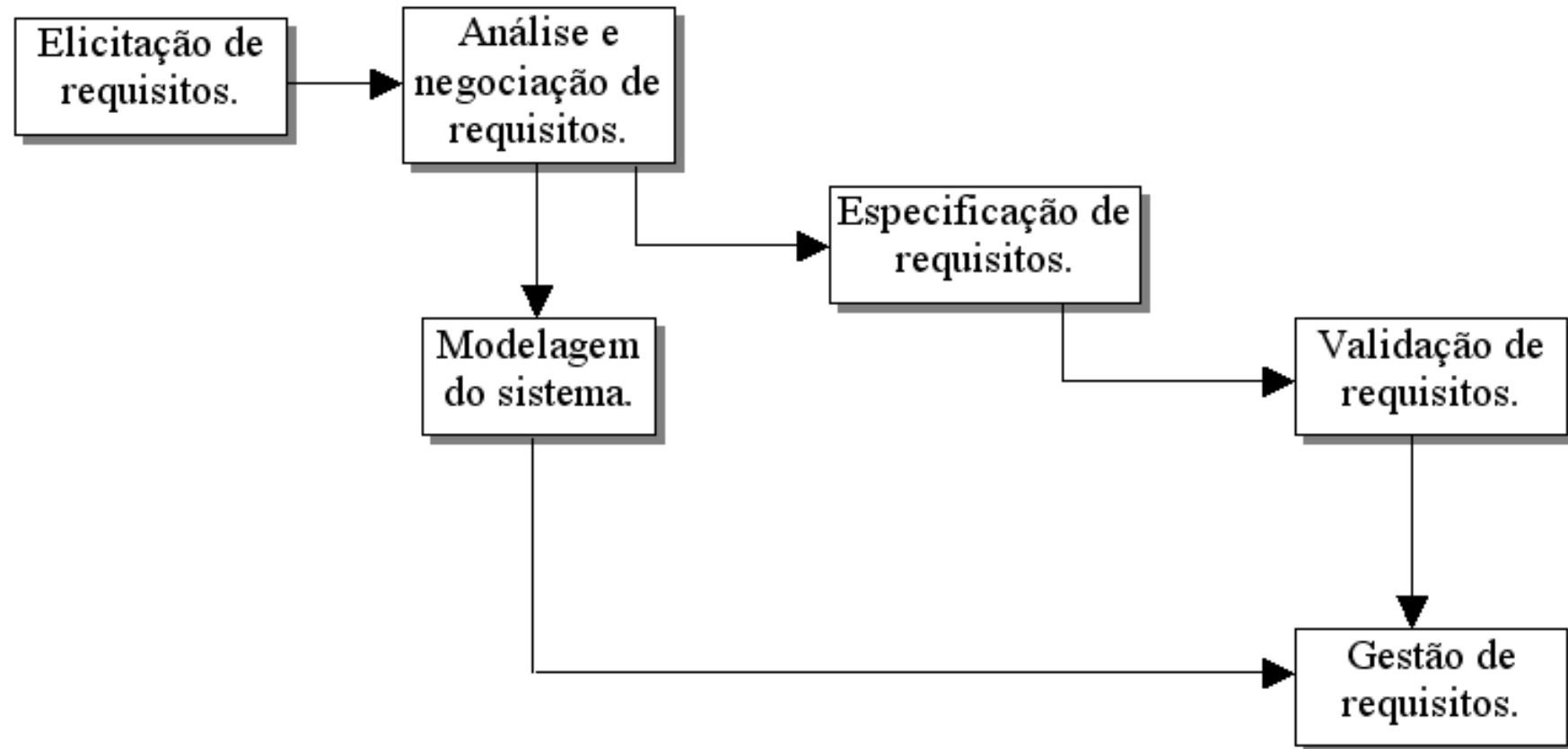


Figura 1 – Processos da engenharia de requisitos.

# Fundamentação

- Gerenciamento de Requisitos: controla as alterações nos requisitos do sistema.
  - Rastreabilidade: mostra o impacto das alterações sobre o sistema. Na tabela de rastreamento é possível visualizar o relacionamento entre requisitos.

# Fundamentação

- IEEE-830-1998: apresenta as partes essenciais que um documento de ERS de boa qualidade deve possuir.



# Fundamentação

## **1. Introdução**

*1.1 Propósito*

*1.2 Escopo*

*1.3 Definições, acrônimos e abreviatura*

*1.4 Referências*

*1.5 Visão Geral*

## **2. Descrição Geral**

*2.1 Perspectiva do produto*

*2.2 Funções do produto*

*2.3 Características do usuário*

*2.4 Restrições*

*2.5 Dependências e suposições*

## **3. Requisitos Específicos**

**Apêndices**

**Índice**

Figura 2 – Seções do documento de ERS.

# Fundamentação

- I-CASE: conjunto de ferramentas com interface padronizada operando sobre um único repositório de dados.
  - Benefícios:
    - subsistemas específicos por finalidade no gerenciamento da informação;
    - informação em tempo real entre os subsistemas.

# Fundamentação

- Enterprise Architect: ferramenta CASE baseada na UML, desenvolvida pela Sparx Systems.
  - Recursos:
    - abrange todas as etapas de especificação de um projeto de software;
    - disponibiliza a interface de automação.

# Fundamentação

- Gerenciamento de requisitos no EA.
  - Restrições:
    - o documento de ERS não é baseado no modelo IEEE-830-1998;
    - limita a leitura do texto do requisito na matriz de rastreabilidade;
    - não indica na matriz o requisito alterado;
    - não ordena automaticamente o identificador único do requisito.

# Fundamentação

- Estrutura e Interface de Automação com EA.
  - Conceito: disponibilizar uma biblioteca para que outras ferramentas acessem e gerenciem seus elementos, este recurso vai ao encontro do conceito de I-CASE.

# Fundamentação

```
procedure TRequisitosEspecificos.CarregaRequisitosEA(const p_repositorioEA : TRepository);  
var i : Integer;  
    l_u : IUnknown;  
    l_p : IDualPackage;  
begin  
    FArvoreRequisitos.Items.Item[c_noPai].DeleteChildren;  
    try  
        for i:=0 to Pred(p_repositorioEA.Models.Count) do begin  
            l_u := p_repositorioEA.Models.GetAt(i);  
            l_u.QueryInterface(IID_IDualPackage, l_p);  
  
            GeraArvore(FnoPai, l_p);  
        end;  
    finally  
        FArvoreRequisitos.FullCollapse;  
        FArvoreRequisitos.Refresh;  
    end;  
end;
```

Figura 3 – Obtendo o elemento modelo do EA

# Projeto

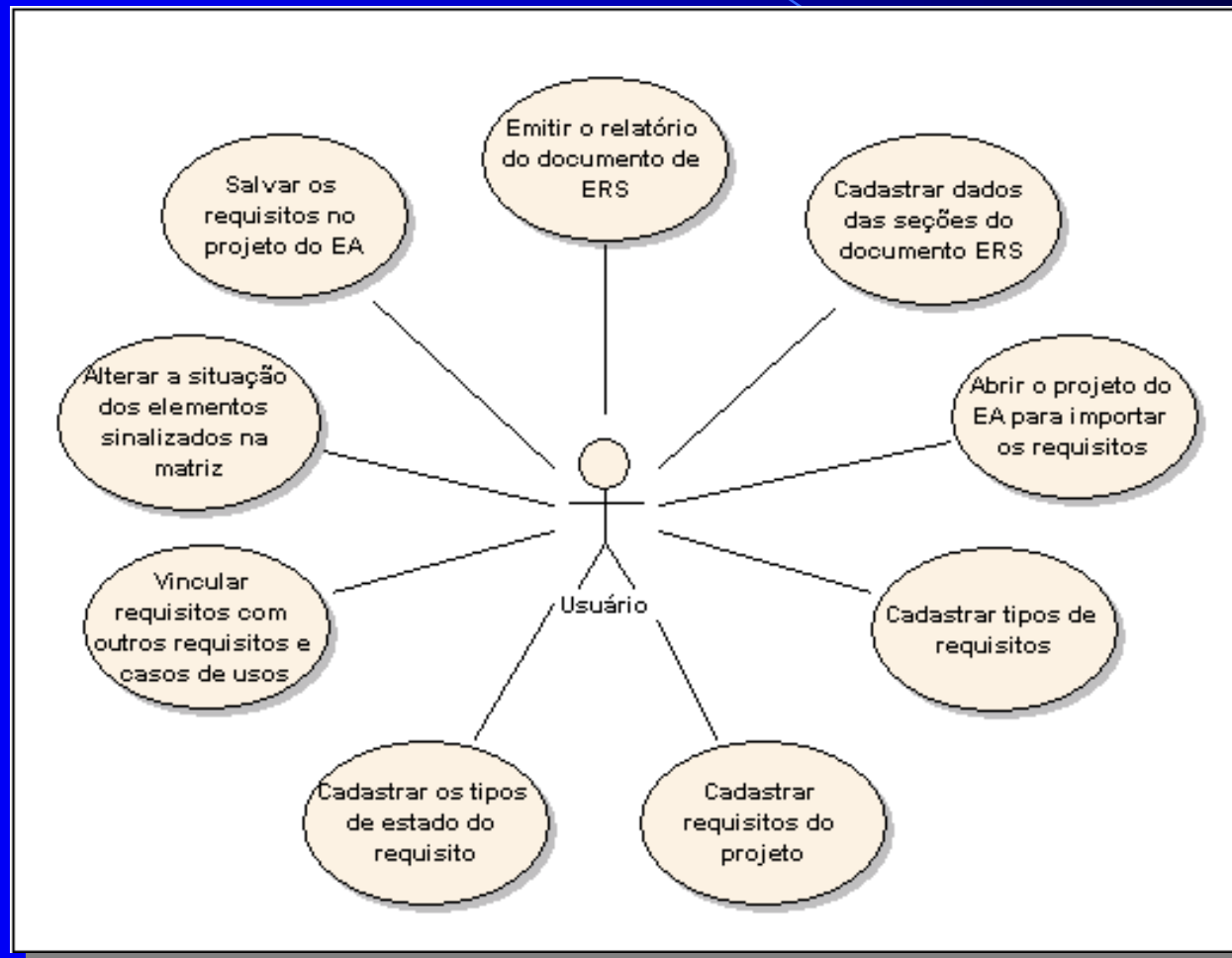


Figura 4 – Diagrama de casos de uso

# Projeto

- Requisitos Não Funcionais:
  - Documento de ERS em conformidade com o modelo IEEE-830-1998;
  - Utilizar técnica de “Estrutura de Árvore”;
  - Portabilidade com Windows 2000/XP;
  - Controles ActiveX para acessar a interface do EA e manipular seus objetos;
  - Ícones representativos para identificar a situação do requisito na matriz.



# Projeto

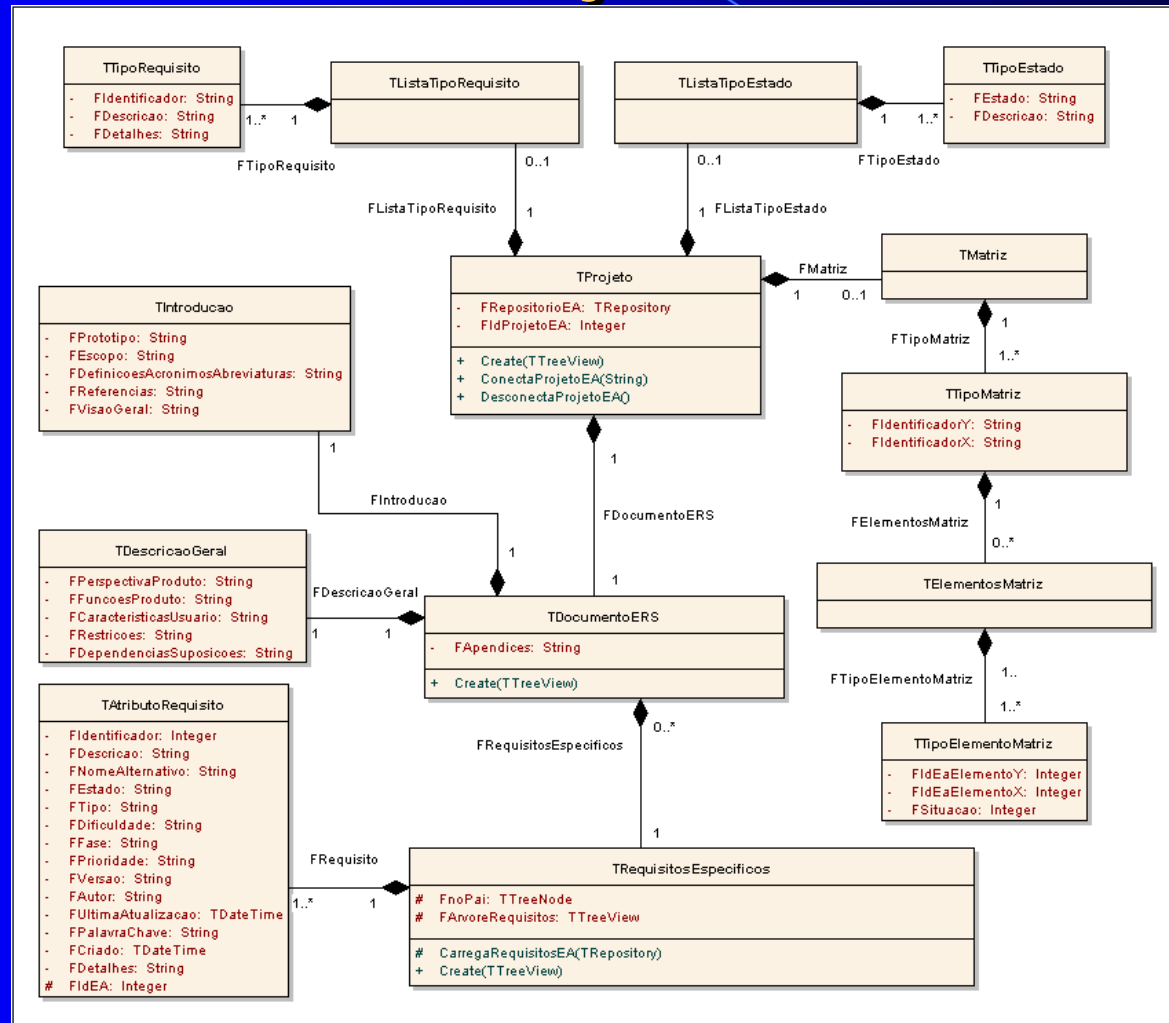


Figura 5 – Diagrama de Classes.

# Projeto

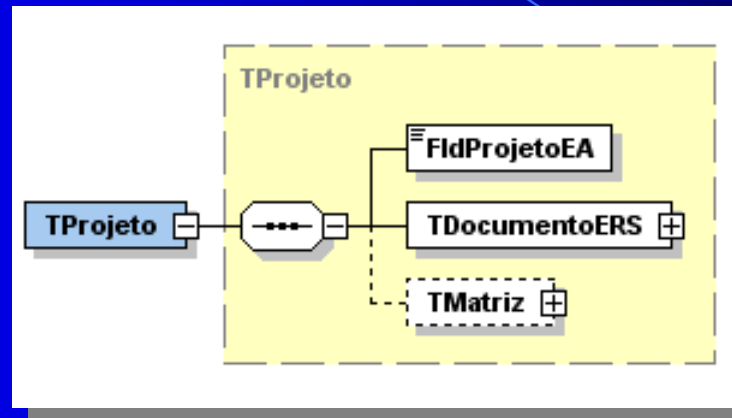


Figura 6 – XML Schema do projeto.

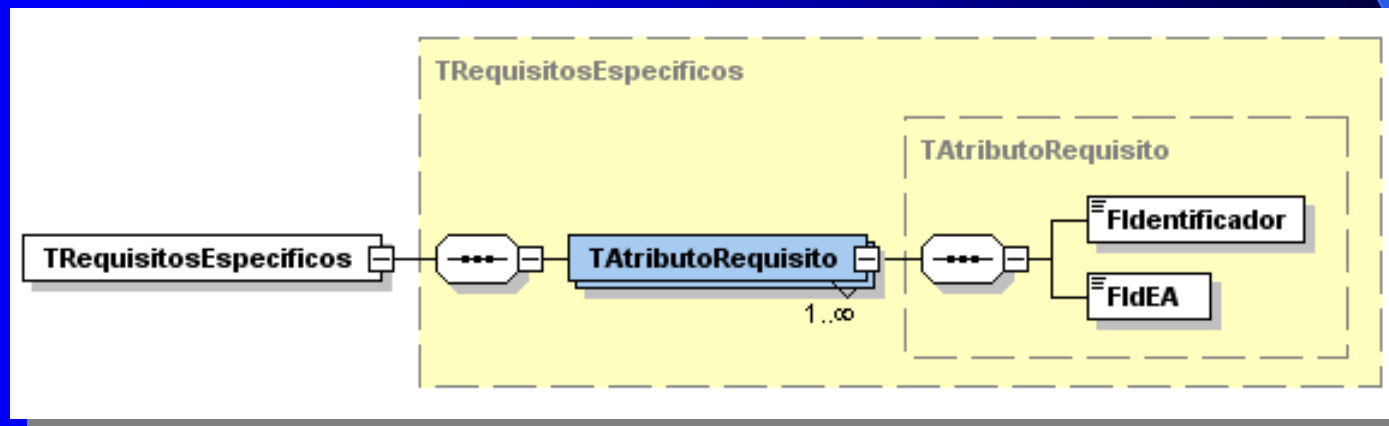


Figura 7 – XML Schema dos requisitos específicos.

# Operacionalidade

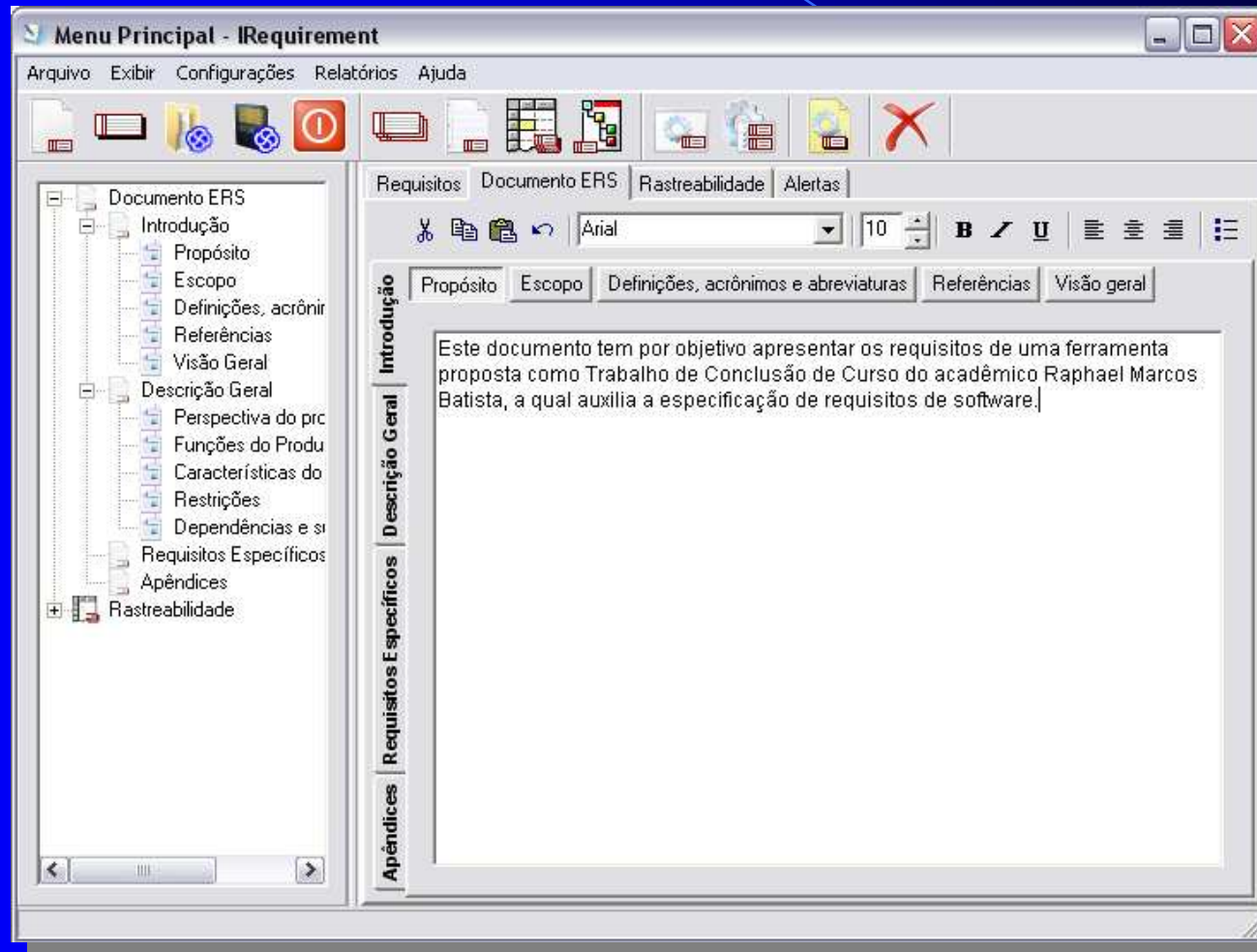


Figura 8 – Sessões do documento de ERS.

# Trabalhos Correlatos

Ferramenta \ Característica	Documentação de Requisitos	Requisitmanager	SPRES	RaQuest	IRequirement
Rastreabilidade dos Requisitos	Não	Sim	Sim	Sim	Sim
Custo	Não especificado	Não especificado	Não especificado	US\$ 150,00	Não especificado
Padrão do documento de ERS	Próprio	IEEE-830-1993	IEEE-830-1998 e RUP	Próprio	IEEE-830-1998
Plataforma	Windows	Web	Windows	Windows	Windows
Integração com Ferramentas CASE	Não	Não	Não	Sim	Sim

Figura 5 – Comparativo entre as ferramentas

# Trabalhos Correlatos

- Flexibilidade da configuração da matriz de rastreabilidade, utilizando requisitos e casos de uso.
- Sessões do documento de ERS de acordo com o modelo IEEE-830-1998.
- Integração com o EA.
- Interface própria para o gerenciamento de requisitos.
- Mensagens de Alerta.

# Conclusões

- Delphi 6 com Active X.
- Conceito de I-CASE no EA.
- EA possui pouca flexibilidade no gerenciamento de requisitos.
- Possibilidade de salvar o documento de ERS em formato RTF.
- Documento de ERS em conformidade com o padrão IEEE-830-1998.

# Conclusões

- Recursos adicionais utilizando XML.
- Gerenciamento de mudanças com a matriz de rastreabilidade com base nos indicadores.

# Extensões

- segurança dos usuários da ferramenta através de controle de acesso e histórico de alterações.
- fórum de discussão, para cada requisito cadastrado, o usuário poderá postar seus questionamentos e sugestões sobre o objetivo e finalidade do qual o requisito se destina.