



ENGENHARIA DE REQUISITOS

Sandra Fabbri



Sumário

- Requisitos Funcionais, Não-Funcionais e do Domínio
- Requisitos do Usuário
- Requisitos do Sistema
- O Documento de Requisitos do software

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos de Software

- Descrições e especificações de um sistema

Sandra Fabbri – sfabbri@dc.ufscar.br

Engenharia de Requisitos

- É o processo de estabelecer os serviços que o cliente requer do sistema e as restrições sob as quais ele opera e é desenvolvido.
- Os Requisitos são as descrições dos serviços do sistema e as restrições que são geradas durante o processo de engenharia de requisitos.

Sandra Fabbri – sfabbri@dc.ufscar.br

O que é um requisito?

- É uma declaração de um serviço ou restrição do sistema que pode variar de um alto nível de abstração até uma especificação funcional detalhada matematicamente.
- O Requisito pode servir para uma dupla função:
 - Pode ser a base de uma oferta de contrato – portanto ele deve estar aberto a interpretações
 - Pode ser a base de um contrato por si só – portanto deve estar definido em detalhes

Sandra Fabbri – sfabbri@dc.ufscar.br

Abstração dos Requisitos (Davis)

“Se uma companhia deseja fazer um contrato para um grande projeto de desenvolvimento de software, ela deve definir as suas necessidades de uma maneira suficientemente abstrata de tal forma que a solução não fique pré-definida. Os requisitos devem estar escritos de tal forma que os contratantes possam fazer um lance de contrato, oferecendo talvez, diferentes maneiras de atender as necessidades das organizações clientes. Uma vez que o contrato tenha sido acertado, o contratado deve escrever uma descrição mais detalhada do sistema para o cliente de modo que o cliente possa compreender e validar o que o software irá fazer. Ambos os documentos podem ser chamados de Documento de Requisitos do sistema.”

Sandra Fabbri – sfabbri@dc.ufscar.br

Tipos de requisitos

- Requisitos do Usuário
 - Declarações em linguagem natural mais diagramas dos serviços fornecidos pelo sistema e suas restrições operacionais. Escrito para os clientes.
- Requisitos do Sistema
 - Um documento estruturado mostrando descrições detalhadas dos serviços do sistema. Escrito como um contrato entre cliente e contratado.
- Especificação do Software
 - Uma descrição detalhada do software que pode servir de base para o projeto ou implementação. Escrita para os desenvolvedores.

Sandra Fabbri – sfabbri@dc.ufscar.br

Definições e Especificações

Definição dos Requisitos do Usuário

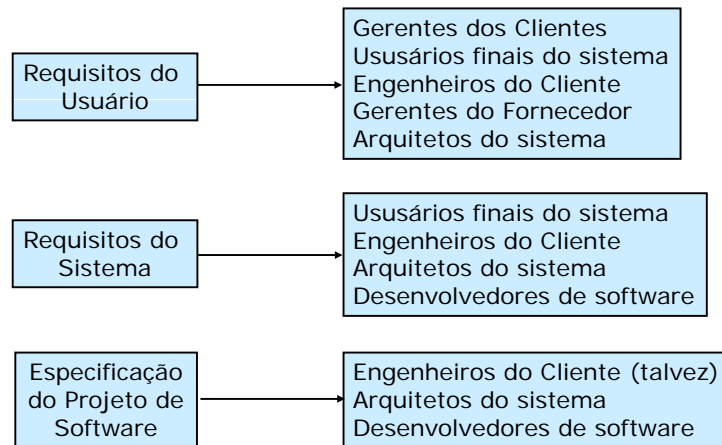
1. O software deve fornecer um meio de representar e acessar arquivos externos criados por outras ferramentas

Especificação dos Requisitos do Sistema

1. O usuário deve dispor de recursos para definir o tipo dos arquivos externos
2. Cada tipo de arquivo externo pode ter uma ferramenta associada que pode ser aplicada a ele
3. Cada tipo de arquivo externo pode ser representado como um ícone específico na tela do usuário
4. Devem ser fornecidos recursos para o ícone que representa um arquivo externo, a ser definido pelo usuário
5. Quando o 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

Sandra Fabbri – sfabbri@dc.ufscar.br

Leitores dos Requisitos



Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos Funcionais/ Não-funcionais/ do Domínio

- **Requisitos Funcionais**
 - Declarações de funções que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações. Eles podem também declarar o que o sistema não deve fazer.
- **Requisitos Não-Funcionais**
 - Restrições dos serviços ou funções fornecidas pelo sistema tais como restrições de tempo, restrições sobre o processo de desenvolvimento, padrões, etc.
- **Requisitos do Domínio**
 - Requisitos que são próprios do domínio da aplicação e que refletem características desse domínio. Podem ser requisitos funcionais ou não-funcionais.

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos Funcionais

- Descrevem a funcionalidade ou os serviços do sistema
- Dependem do tipo do software, usuários esperados e do tipo do sistema em que o software será usado
- Requisitos Funcionais do Usuário podem ser declarações em alto nível sobre o que o sistema deve fazer, mas os Requisitos Funcionais do Sistema devem descrever os serviços do sistema em detalhes

Sandra Fabbri – sfabbri@dc.ufscar.br

Exemplos de Requisitos Funcionais

- O usuário deve conseguir procurar todo o conjunto inicial de banco de dados ou selecionar um subconjunto dele.
- O sistema deve fornecer telas apropriadas para o usuário ler documentos no repositório de documentos.
- A todo pedido deve ser alocado um identificador único (ORDER_ID) o qual poderá ser copiado para a área de armazenagem permanente da conta.

Sandra Fabbri – sfabbri@dc.ufscar.br

Qualidades dos Requisitos - precisão

- Surgem vários problemas quando os requisitos não são declarados de forma precisa.
- Requisitos ambíguos podem ser interpretados de diferentes maneiras pelos desenvolvedores e usuários.
- Considere o termo 'telas apropriadas'
 - Intenção do Usuário – telas especiais para cada tipo diferente de documento
 - Interpretação do Desenvolvedor – fornecer uma tela texto que mostra o conteúdo do documento

Sandra Fabbri – sfabbri@dc.ufscar.br

Qualidades dos Requisitos - completeza e consistência

- Os requisitos devem ser completos e consistentes
 - Completo
 - Eles devem incluir descrição de todas as facilidades que estão sendo requeridas
 - Consistente
 - Eles não devem apresentar conflitos ou contradições entre as descrições das facilidades fornecidas pelo sistema
- Na prática, é impossível produzir um Documento de Requisitos completo e consistente

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos Não-Funcionais

- Definem propriedades e restrições do sistema, como por exemplo, confiabilidade, tempo de resposta e requisitos de armazenamento. Restrições são capacidade do dispositivo de E/S, representações do sistema, etc.
- Podem também ser requisitos associados ao processo, como a especificação de um padrão de qualidade, o uso de algumas ferramentas CASE para gerar alguns documentos, etc.
- Requisitos Não-Funcionais podem ser mais críticos que os Requisitos Funcionais, isto é, se eles não forem atendidos, o sistema pode ser inútil.

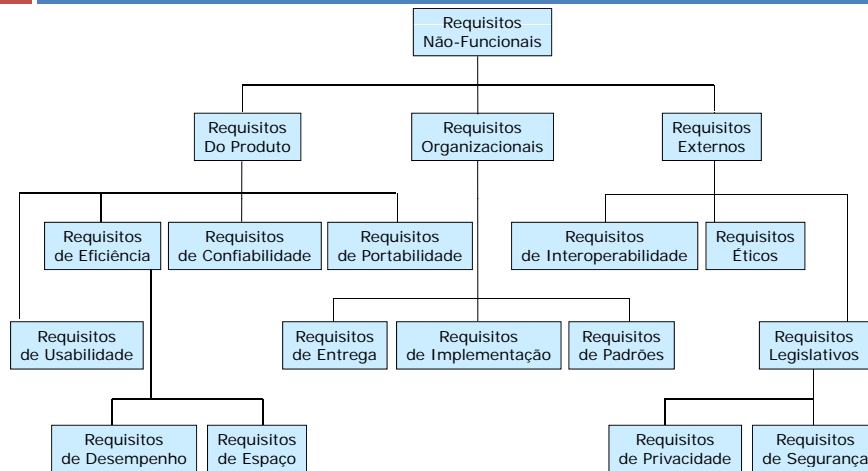
Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos Não-Funcionais: classificação

- Requisitos do Produto
 - São aqueles que especificam que o produto deve se comportar de uma determinada maneira, por ex., velocidade de execução, confiabilidade, etc.
- Requisitos Organizacionais
 - São aqueles que são consequência da política e dos procedimentos organizacionais, por ex., padrões de processo usados, requisitos de implementação, etc.
- Requisitos Externos
 - São aqueles que surgem de fatores externos ao sistema e ao seu processo de desenvolvimento, como por ex., requisitos de interoperabilidade, requisitos legislativos, etc.

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos Não-Funcionais: tipos



Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos Não-Funcionais: exemplos

- **Requisitos do Produto**
 - Toda comunicação necessária entre o SistemaY e o usuário deve ser expressa usando o conjunto padrão de caracteres XXX.
- **Requisitos Organizacionais**
 - O processo de desenvolvimento do sistema e os produtos liberáveis devem estar em conformidade com o processo e os liberáveis definidos em XYZCo-SP-STAN-04
- **Requisitos Externos**
 - Os operadores do sistema não devem ter acesso a qualquer dado que não necessitem

Sandra Fabbri – sfabbri@dc.ufscar.br

Metas e Requisitos Verificáveis

- **Requisitos Não-Funcionais** podem ser muito difíceis de serem declarados precisamente e requisitos imprecisos podem ser difíceis de serem verificados.
- **Metas**
 - São úteis aos desenvolvedores uma vez que elas transmitem as intenções dos usuários do sistema
 - Ex: uma intenção genérica do usuário, como “fácil de ser usado”
- **Requisito Não-Funcional Verificável**
 - É uma declaração que usa alguma medida que possa ser objetivamente testada

Sandra Fabbri – sfabbri@dc.ufscar.br

Exemplos

- **Meta do Sistema**
 - O sistema deve ser fácil de ser usado por controladores experientes e deve estar organizado de tal maneira que os erros dos usuários sejam minimizados.
- **Requisito Não-Funcional Verificável**
 - Controladores experientes devem ser capazes de usar todas as funções do sistema depois de duas horas de treinamento. Depois desse treinamento, o número médio de erros feito por um usuário experiente não deve exceder dois erros por dia.

Sandra Fabbri – sfabbri@dc.ufscar.br

Métricas para especificar Requisitos Não-Funcionais

| Propriedade | Métrica |
|-------------------|--|
| Velocidade | Transações processadas/segundo Tempo de resposta ao usuário/evento Tempo de refresh da tela |
| Tamanho | K Bytes Tamanho específico de memória |
| Facilidade de uso | Tempo de treinamento Número de frames de Help |
| Confiabilidade | Tempo médio de falha Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade |
| Robustez | Tempo de reinício depois de falha Porcentagem de eventos que causam falhas Probabilidade de que dados sejam corrompidos por falhas |
| Portabilidade | Porcentagem de declarações dependentes do sistema alvo Número de sistemas alvo |

Sandra Fabbri – sfabbri@dc.ufscar.br

Interação entre Requisitos

- Conflitos entre diferentes Requisitos Não-Funcionais são comuns em sistemas complexos
- Sistema para Espaçonaves
 - Para minimizar o peso, o número de chips do sistema deve ser minimizado
 - Para minimizar o consumo de energia, chips de menor potência devem ser usados
 - Entretanto, usar chips de menor potência pode significar que mais chips devem ser usados. Qual é o requisito mais crítico?

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Domínio

- São derivados do domínio da aplicação e descrevem características e aspectos do sistema que refletem o domínio
- Podem ser novos Requisitos Funcionais, restrições sobre os requisitos existentes ou definir computações específicas
- Se os Requisitos do Domínio não são satisfeitos, pode não ser possível operar o sistema de forma efetiva

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Domínio: exemplo – sistema de biblioteca

- Deve haver uma interface-padrão com o usuário para todas as bases de dados, a qual deve estar baseada no padrão Z39.50.
- Devido a restrições de direitos autorais, alguns documentos devem ser eliminados imediatamente à sua chegada. Dependendo dos requisitos do usuário, esses documentos serão impressos localmente no servidor do sistema para serem encaminhados manualmente ao usuário ou direcionados para uma impressora em rede.

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Domínio: exemplo – sistema de proteção de trem

- A desaceleração do trem deve ser calculada como:
 - ▣ $D_{trem} = D_{controle} + D_{gradiente}$
- onde $D_{gradiente}$ é
- $9.81ms^2 * \text{gradiente compensado}/\alpha$
- e onde os valores de $9.81ms^2 / \alpha$ são conhecidos para diferentes tipos de trem.

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Domínio - problemas

- Dificuldade de compreensão
 - ▣ Os Requisitos são expressos em uma linguagem própria do Domínio da Aplicação
 - ▣ Em geral, essa linguagem não é compreendida pelos engenheiros de software
- Conhecimento tácito
 - ▣ Os especialistas no Domínio entendem tão bem da área que não consideram necessário explicitar os Requisitos do Domínio

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Usuário

- Devem descrever os requisitos Funcionais e Não-Funcionais de modo compreensível pelos usuários do sistema, os quais não têm conhecimento técnico detalhado
- São definidos usando linguagem natural, tabelas e diagramas

Sandra Fabbri – sfabbri@dc.ufscar.br

Problemas com a definição dos Requisitos do Usuário

- Falta de clareza
 - ▣ É difícil utilizar a linguagem de maneira precisa e não ambígua sem produzir um documento difícil de ler
- Confusão de Requisitos
 - ▣ Os requisitos Funcionais e 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

Sandra Fabbri – sfabbri@dc.ufscar.br

Guidelines para escrever Requisitos do Usuário

- ❑ Invente um formato padrão e use-o em todos os requisitos
- ❑ Use a linguagem de forma consistente. Use “deve” para requisitos obrigatórios e “deveria” para requisitos desejáveis, isto é, requisitos que sejam feitos preferencialmente como foram declarados, mas se necessário, por algum motivo justificável, podem ser feitos de outra forma.
- ❑ Utilize destaques no texto para identificar partes importantes dos requisitos.
- ❑ Evite o uso de jargões de informática.

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Sistema

- ❑ São especificações mais detalhadas dos Requisitos dos Usuários
- ❑ Servem como base para projetar o sistema
- ❑ Podem ser usados como parte do contrato do sistema
- ❑ Podem ser expressos usando modelos específicos (modelo de dados, modelo comportamental, etc.)
- ❑ Muitas vezes utiliza-se a Linguagem Natural para especificar os Requisitos de Sistema

Sandra Fabbri – sfabbri@dc.ufscar.br

Requisitos do Sistema e Projeto

- ❑ Em princípio, os **requisitos** deveriam declarar **o que** o sistema deveria fazer e o **projeto** deveria descrever **como** ele teria que ser implementado
- ❑ Na prática, os requisitos e o projeto são inseparáveis. As razões para isso são:
 - A arquitetura do sistema pode ser projetada para ajudar a estruturar a especificação de requisitos
 - O sistema pode interoperar com outros sistemas, o que pode gerar requisitos de projeto
 - O uso de um projeto específico pode ser um requisito externo de sistema

Sandra Fabbri – sfabbri@dc.ufscar.br

Problemas com a Especificação em Linguagem Natural

- ❑ Ambigüidade
 - Tanto leitores como redatores de requisitos devem interpretar as mesmas palavras da mesma forma. A LN é naturalmente ambígua, de forma que isso é muito difícil.
- ❑ Muita Flexibilidade
 - A mesma coisa pode ser dita de inúmeras maneiras na especificação.
- ❑ Falta de Modularização
 - As estruturas da LN são inadequadas para estruturar os requisitos do sistema.

Sandra Fabbri – sfabbri@dc.ufscar.br

Alternativas para a LN

| Notação | Descrição |
|-----------------------------------|--|
| Linguagem natural estruturada | Depende da definição de formulários-padrão ou templates para especificar os requisitos do sistema. |
| Linguagem de descrição de projeto | Essa abordagem usa uma linguagem parecida com linguagem de programação, mas que possui recursos mais abstratos para especificar os requisitos do sistema. |
| Notação gráfica | Linguagem gráfica complementada com anotações textuais usada para descrever requisitos funcionais do sistema. Exemplo: SADT (Ross, 1977; Schoman and Ross, 1977) Mais recentemente: descrição de Casos de Uso (Jacobsen, Christerson et al., 1993) |
| Especificações Matemáticas | São notações que usam uma base matemática Ex: Máquinas de Estados Finitos Não são ambíguas, mas são difíceis de serem entendidas pelo usuário e também para servirem como um contrato. |

Sandra Fabbri – sfabbri@dc.ufscar.br

Especificação em Linguagem Estruturada

- Formas limitadas da LN podem ser usadas para expressar os requisitos
- Como é limitada, remove alguns problemas de ambigüidade e flexibilidade e impõe um grau de uniformidade na especificação
- Geralmente é apoiada por uma abordagem baseada em formulários

Sandra Fabbri – sfabbri@dc.ufscar.br

Especificação em Linguagem Estruturada - Formulário

- Definição das entidades e funções
- Descrição das entradas e de onde elas vêm
- Descrição das saídas e para onde elas vão
- Indicação de outras entidades requeridas
- Pré e pós condições (se for apropriado)
- Descrição dos efeitos colaterais da operação (caso existam)

Sandra Fabbri – sfabbri@dc.ufscar.br

Exemplo: especificação baseada em formulário

| | |
|-----------------------------|--|
| Cadastamento de Funcionário | |
| Função | Cadasta Funcionário |
| Descrição | Essa função tem o objetivo de inserir os dados de um funcionário no banco de dados |
| Entrada | Nome, endereço, telefone, |
| Fonte | Todos os dados são informados pelo funcionário. |
| Saída | Registro no banco de dados. |
| Destino | Banco de dados |
| Pre-condição | O funcionário já deve possuir um código de identificação. |
| Post-condition | |
| Efeitos-colaterais | Nenhum |
| ... | |

Sandra Fabbri – sfabbri@dc.ufscar.br

Especificação em PDL

- Os Requisitos podem ser definidos operacionalmente, usando uma linguagem do tipo de uma linguagem de programação, mas com mais flexibilidade de expressão
- Seu uso é mais apropriado em duas situações
 - Onde uma operação é especificada como uma seqüência de ações em que a ordem é importante
 - Quando interfaces de hardware e software devem ser especificados

Sandra Fabbri – sfabbri@dc.ufscar.br

Desvantagens da PDL

- A PDL pode não ser suficientemente expressiva para definir conceitos do Domínio e expressar as funcionalidades de uma forma compreensível
- A notação só é compreensível às pessoas que têm conhecimento de linguagem de programação
- Os Requisitos podem ser vistos como uma especificação de projeto ao invés de um modelo que ajude a compreensão do sistema

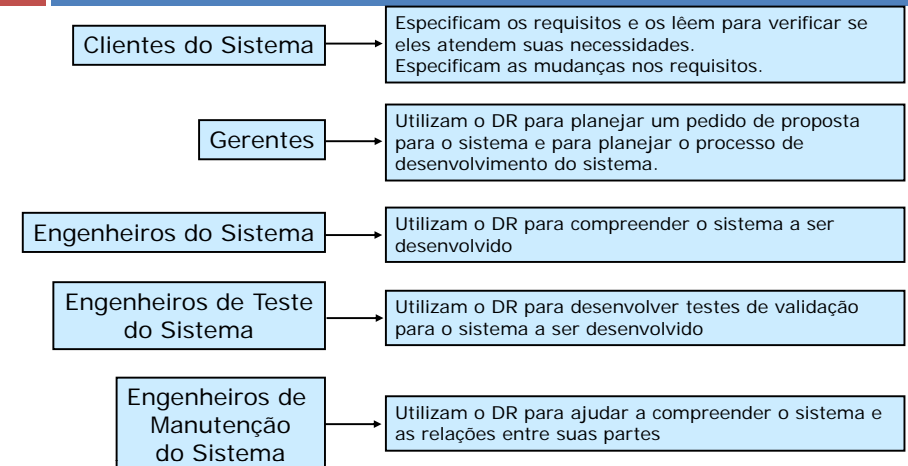
Sandra Fabbri – sfabbri@dc.ufscar.br

Documento de Requisitos

- O Documento de Requisitos (DR) é a declaração oficial do “o que” que é solicitado aos desenvolvedores do sistema
- O DR deve incluir tanto a **definição** (requisitos do usuário) como a **especificação** (requisitos do sistema) dos requisitos
- O DR **NÃO** é um documento de projeto. Na medida do possível ele deve declarar “**O QUE**” o sistema deve fazer e **não** “**COMO**” deve ser feito

Sandra Fabbri – sfabbri@dc.ufscar.br

Usuários do Documento de Requisitos



Sandra Fabbri – sfabbri@dc.ufscar.br

Características requeridas em um Documento de Requisitos

- ❑ Especificar o comportamento externo do sistema
- ❑ Especificar restrições de implementação
- ❑ Ser fácil de alterar
- ❑ Servir como um guia de referência para atividades de manutenção
- ❑ Registrar precauções relacionadas ao ciclo de vida do sistema, isto é, predizer alterações (análise de risco e medidas de contingência)
- ❑ Caracterizar as respostas aos eventos inesperados

Sandra Fabbri – sfabbri@dc.ufscar.br

Padrão IEEE

- ❑ Introdução
- ❑ Descrição Geral
- ❑ Requisitos específicos
- ❑ Apêndices
- ❑ Índice

- ❑ Essa estrutura é genérica e deve ser instanciada para cada sistema

Sandra Fabbri – sfabbri@dc.ufscar.br

Estrutura do Documento de Requisitos (mais detalhada)

- ❑ Introdução
- ❑ Glossário
- ❑ Definição dos Requisitos do Usuário
- ❑ Arquitetura do Sistema
- ❑ Especificação dos Requisitos do Sistema
- ❑ Modelos do Sistema
- ❑ Evolução do Sistema
- ❑ Apêndices
- ❑ Índice

Sandra Fabbri – sfabbri@dc.ufscar.br

Pontos Chave

- ❑ Os Requisitos devem declarar o que o sistema deve fazer e definir as restrições relacionadas às suas operações e à implementação.
- ❑ Os Requisitos Funcionais devem declarar os serviços que o sistema deve fornecer.
- ❑ Os Requisitos Não-Funcionais restringem o sistema que está sendo desenvolvido ou o processo de desenvolvimento.
- ❑ Os Requisitos do Usuário são declarações em alto-nível do que o sistema deve fazer.

Sandra Fabbri – sfabbri@dc.ufscar.br

Pontos Chave

- ❑ Os Requisitos do Usuário devem ser escritos em Linguagem Natural, Tabelas e Diagramas.
- ❑ Os Requisitos do Sistema têm por objetivo explicar as funções que o sistema deve fornecer.
- ❑ Os Requisitos do Sistema podem ser escritos em Linguagem Natural, PDL ou em Linguagem Formal.
- ❑ O Documento de Requisitos do software é uma declaração de concordância em relação aos requisitos definidos para o sistema.