

Requisitos de Software

Eduardo Figueiredo

<http://www.dcc.ufmg.br/~figueiredo>

dcc603@gmail.com

26 Março 2012

Tópicos da Aula

- Requisitos
 - Funcionais e não funcionais
 - De usuário e do sistema
- Engenharia de requisitos
 - Estudo de viabilidade
 - Elicitação e análise de requisitos
 - Validação dos requisitos

Que é são requisitos?

- Descrições
 - De serviços fornecidos pelo sistema
 - De restrições destes serviços
- Podem ser especificado formalmente
- Engenharia de Requisitos
 - Processo de descobrir, analisar, documentar e verificar requisitos

Tipos de Requisitos

- Classificação
 - Requisitos Funcionais
 - Requisitos Não Funcionais
- Visão dos requisitos
 - Requisitos de Usuário
 - Requisitos do Sistema

Classificação de Requisitos

Requisitos Funcionais
Requisitos Não Funcionais

Requisitos Funcionais

- Descrevem explicitamente as funcionalidades e serviços do sistema
- Documenta
 - como o sistema deve reagir a entradas específicas
 - como deve se comportar em determinadas situações
 - o que o sistema não deve fazer

Atributos dos Requisitos Funcionais

- **Completezude**
 - Todas os serviços devem estar definidos
- **Consistência**
 - Os requisitos não devem ter definições contraditórias
- Na prática é quase impossível atingir completude e consistência dos requisitos

Exemplos de Requisitos Funcionais

- O usuário pode pesquisar todo ou um sub-conjunto do banco de dados
- O sistema deve oferecer telas apropriadas para o usuário ler documentos armazenados
- Cada pedido deve ser associado a um identificador único (PID), o qual o usuário pode copiar para a área de armazenamento permanente da conta

Ambiguidades em Requisitos

- A imprecisão na especificação de requisitos é motivo de vários problemas
 - O desenvolvedor tende a interpretar o requisito da maneira mais fácil de implementar
- “O sistema deve oferecer telas apropriadas...”
 - O que são telas apropriadas?

Requisitos Não-Funcionais (RNF)

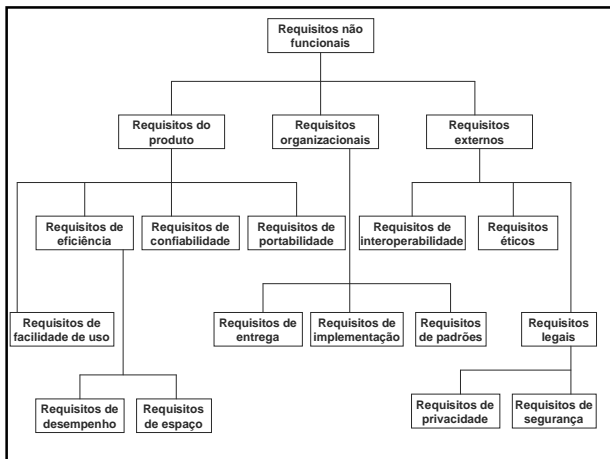
- Definem propriedades e restrições do sistema
 - Exemplos: segurança, desempenho, espaço em disco
- Podem ser do sistema todo ou de partes do sistema
- Requisitos não-funcionais podem ser mais críticos que requisitos funcionais
 - Se não satisfaz, o sistema é inútil

Classificação de RNF

- **Requisitos do Produto**
 - Especificam o comportamento do software (ex.: desempenho)
- **Requisitos Organizacionais**
 - Consequência de políticas e procedimentos das empresas (ex.: padrões do cliente)
- **Requisitos Externos**
 - Derivados do ambiente ou fatores externos ao sistema (ex.: legislação)

Exemplos de RNF

- **Requisitos do Produto**
 - A interface do usuário deve ser implementada como simples HTML
- **Requisitos Organizacionais**
 - Todos os documentos entregues devem seguir o padrão de relatórios XYZ-00
- **Requisitos Externos**
 - Informações pessoais do usuário não podem ser vistas pelos operadores do sistema



Verificação de RNF

- Às vezes são de difícil verificação
- Idealmente, requisitos não-funcionais devem ser mensuráveis
 - Após a implementação, estes podem ser testados objetivamente

Métricas de RNF

- Velocidade
 - Transações processadas por segundo
 - Tempo de resposta
 - Tempo de atualização de tela
- Facilidade de uso
 - Tempo gasto em treinamento
 - Número de frames de ajuda

Métricas de RNF

- Confiabilidade
 - Tempo médio para falhar
 - Probabilidade de indisponibilidade
 - Taxa de ocorrência de falhas
 - Disponibilidade
- Robustez
 - Tempo de reinício após uma falha
 - Porcentagem de eventos que causam falhas

Alguns Problemas de RNF

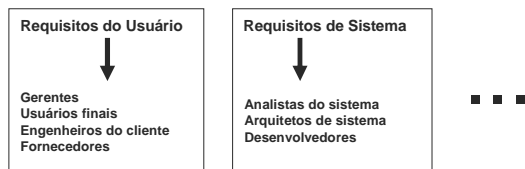
- A especificação quantitativa de requisitos não funcionais é difícil
- Ocorre mistura de requisitos funcionais e não funcionais
- Requisitos não funcionais podem conflitar com outros requisitos (funcionais ou não)

Visões de Requisitos

Requisitos de Usuário
Requisitos do Sistema

Visões da Especificação

- É interessante gerar diferentes visões da especificação de sistema
 - Comunicação com diferentes tipos de interessados (*stakeholders*)



Requisitos de Usuário

- Descreve as funções e restrições do sistema de forma abstrata
 - Inteligível pelo usuário / cliente
- Ponto de vista das necessidades da empresa
 - Não indica uma solução
- Escrito em linguagem natural com diagramas simples (ex. tabelas)

Alguns Problemas

- Falta de clareza
 - Pode ser difícil uma linguagem precisa e não ambígua
- Confusão de requisitos
 - Requisitos pode estar misturados com informações do projeto
- Fusão de requisitos
 - Diversos requisitos expressos juntos

Diretrizes Gerais de Redação

- Adotar um formato padrão e usá-lo em todas as definições de requisitos
- Usar a linguagem de forma simples e consistente (pode x deve)
- Usar destaque (negrito, itálico ou sublinhado) para partes importantes
- Evitar usar jargões de informática

Requisitos de sistema

- São descrições mais detalhadas que os requisitos do usuário
- Devem ser padronizados, completos e consistente
 - Usados pela equipe de desenvolvimento
 - Fazem parte do contrato

Exemplo de Problema

- Não é fácil padronizar os requisitos usando linguagem natural
- Na escada rolante dizia
 - “Calçados devem ser usados”
 - “Cachorros devem ser carregados”

Uso de Linguagem Estruturada

- É uma forma restrita da linguagem natural
- Vantagens
 - Mantém a facilidade de expressão e compreensão da linguagem natural
 - Garante algum grau de uniformidade na especificação
 - Podem ser escritas formulários especiais

Requisitos x Projeto

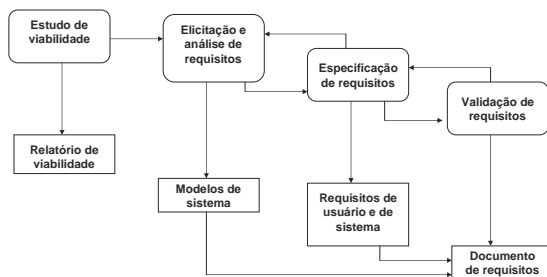
- Requisitos definem **o que** o sistema faz
 - E não **como** ele faz (projeto)
- No entanto, eles sempre incluem, alguns detalhes de projeto, pois...
 - Os requisitos são organizados de acordo com os diferentes subsistemas
 - O sistema deve interoperar com outros sistemas existentes
 - O uso de uma arquitetura específica pode ser um requisito externo do sistema

Engenharia de Requisitos

Eng. de Requisitos: Atividades

- Inclui quatro fases principais
 - Estudo de viabilidade
 - Elicitação (ou análise) de requisitos
 - Especificação de requisitos
 - Validação dos requisitos

Eng. de Requisitos: Processo



O Documento de Requisitos

- Declaração oficial do que os desenvolvedores devem implementar
 - Deve incluir tanto requisitos de usuário quanto requisitos do sistema
- É lido por várias pessoas interessadas no documento (*stakeholders*)

Stakeholders

- Pessoas que têm qualquer influência direta ou indireta sobre os requisitos
 - Clientes e usuários
 - Gerentes de projeto
 - Analistas de sistema
 - Engenheiros de testes
 - Mantenedores, etc.

Estudo de Viabilidade

Estudo de Viabilidade

- Sistemas novos devem começar com um estudo da viabilidade
- Responder Perguntas
 - O sistema contribui para os objetivos gerais da organização?
 - O sistema pode ser implementado com a tecnologia atual dentro das restrições de custo e de prazo?
 - O sistema pode ser integrado a outros sistemas já em operação?

Outras Perguntas...

- Como a organização se comportaria se esse sistema não fosse implementado?
- Quais são os problemas com os processos atuais e como um novo sistema ajudaria a diminuir esses problemas?
- Que contribuição direta o sistema trará para os objetivos da empresa?

Relatório de Viabilidade

- Após responder às perguntas, deve ser preparado o relatório de viabilidade
- O relatório pode
 - Propor mudanças no enfoque, no orçamento e/ou no cronograma
 - Sugerir outros requisitos de alto nível para o sistema
 - Simplesmente cancelar o projeto de desenvolvimento do sistema

Elicitação e Análise de Requisitos

Elicitação e Análise de Requisitos



Descoberta de Requisitos

- O objetivo é descobrir
 - O domínio de aplicação
 - Serviços que devem ser fornecidos pelo sistema
 - Restrições associadas ao domínio ou serviços
- Várias técnicas podem ser usadas
 - Envolvem diversos *stakeholders*

Obtendo os Requisitos

- Técnicas para levantamento de requisitos
 - Descoberta de Requisitos (Pontos de vista)
 - Entrevistas
 - Cenários
 - Casos de Uso
 - Etnografia

Obtendo os Requisitos

- Técnicas para levantamento de requisitos
 - **Descoberta de Requisitos (Pontos de vista)**
 - Entrevistas
 - Cenários
 - Casos de Uso
 - Etnografia

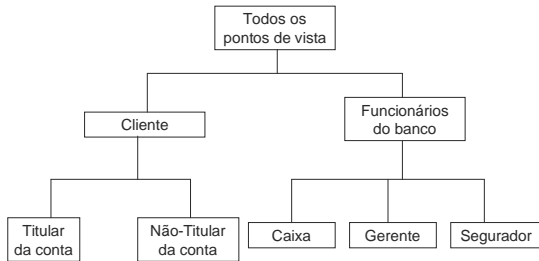
Levantamento por Pontos de Vista

- Em uma empresa de tamanho médio ou grande, existem vários *stakeholders*
- Cada *stakeholder* tem um ponto de vista diferente
 - Cada um vê o problema de modo diferente
- Objetivo: conhecer o problema por várias perspectivas

Pontos de Vista do Banco

Titular da conta	Não-Titular da conta	Caixa do banco
Lista de Serviços Retirar dinheiro Consultar saldo Pedir cheques Enviar mensagem Desbloquear cartão Executar transação Pedir extrato Transferir fundos Contrair empréstimo	Lista de Serviços Retirar dinheiro Consultar saldo	Lista de Serviços Executar diagnósticos Somar dinheiro Colocar papel Enviar mensagem

Estruturar os Pontos de Vista



Algumas Dificuldades

- *Stakeholders* frequentemente:
 - Não sabem na realidade o que querem
 - Não conseguem expressar claramente o que desejam
 - Fazem pedidos não realistas
 - Se expressam com seus próprios termos (técnicos)
- Diferentes *stakeholders* expressam o mesmo requisito de forma diferente

Armadilhas da Técnica

- Alguns *stakeholders* podem pedir requisitos para aumentar o seu poder na empresa
- O ambiente é dinâmico: novos *stakeholders* e novos requisitos podem surgir
- Pontos de vista podem apresentar **duplicidade** ou **inconsistência**

Obtendo os Requisitos

- Técnicas para levantamento de requisitos
 - Descoberta de Requisitos (Pontos de vista)
 - **Entrevistas**
 - Cenários
 - Casos de Uso
 - Etnografia

Entrevistas

- Questionar os stakeholders sobre o sistema (ou processo) atual e sobre o sistema que será desenvolvido
- Tipos de entrevistas
 - Entrevistas fechadas: conjunto pré-definido de perguntas
 - Entrevistas abertas: sem agenda pré-definida; se adapta para explorar o conhecimento do stakeholder

Obtendo os Requisitos

- Técnicas para levantamento de requisitos
 - Descoberta de Requisitos (Pontos de vista)
 - Entrevistas
 - **Cenários**
 - Casos de Uso
 - Etnografia

A Técnica de Cenários

- Descreve uma situação de uso do sistema
- Inclui informações como
 - Nome do Cenário
 - Ator
 - Pré-condição
 - Fluxo normal
 - Fluxos alternativos
 - Pós-condição

Exemplo de Cenário (ATM)

- Nome do Cenário: Sacar dinheiro
- Ator: Correntista
- Pré-condição: Conta e senha validada
- Fluxo normal
 1. Entrar com valor do saque
 2. Confirmar dados e operação
 3. Debitar valor da conta do cliente
- Fluxos alternativo: Saldo insuficiente
 - 3.1 Apresentar aviso ao cliente
- Pós-condição:
 - Valor sacado é debitado do saldo do cliente

Obtendo os Requisitos

- Técnicas para levantamento de requisitos
 - Descoberta de Requisitos (Pontos de vista)
 - Entrevistas
 - Cenários
 - **Casos de Uso**
 - Etnografia

De Cenário ao Caso de Uso

- Casos de Uso identificam
 - Os atores envolvidos
 - As funcionalidades principais
 - A interação entre atores e funcionalidades do sistema

Veremos mais sobre Cenários e Casos de Uso na próxima aula.

Obtendo os Requisitos

- Técnicas para levantamento de requisitos
 - Descoberta de Requisitos (Pontos de vista)
 - Entrevistas
 - Cenários
 - Casos de Uso
 - **Etnografia**

Etnografia

- É uma técnica de observação utilizada para compreender os requisitos sociais e organizacionais
- O analista (engenheiro de requisitos) se insere na organização do cliente
 - Observa o trabalho no dia a dia
 - Anota as tarefas dos funcionários

Etnografia é eficaz...

- Para descobrir como as pessoas realmente trabalham
- Para descobrir a cooperação e conscientização das atividades de outras pessoas
- Para desenvolver um protótipo
- Para descobrir importantes detalhes que outros métodos omitem

Validação dos Requisitos

Objetivos da Validação

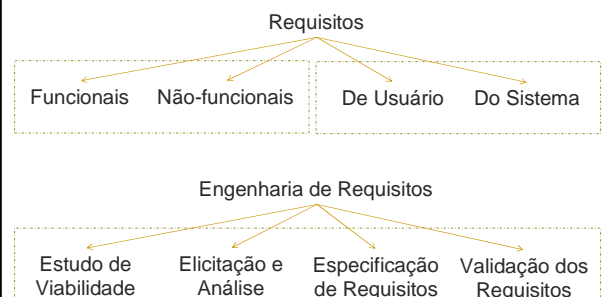
- Mostrar que os requisitos realmente definem o sistema que o cliente deseja
- Descobrir problemas com os requisitos

Verificações da Validação

- Verificações de validade
 - Quais serviços são necessários?
- Verificações de consistência
 - Existe conflitos entre requisitos?
- Verificações de completude
 - Todos os requisitos estão documentados?
- Verificações de realismo
 - Os requisitos podem ser implementados?
- Facilidade de verificação
 - Como verificar se o requisito foi implementado?

Revisão da Aula

Revisão da Aula



[Bibliografia Principal]

- Ian Sommerville. Engenharia de Software, 9ª Edição. Pearson Education, 2011.
 - Cap. 4: Seções 4.1 a 4.6