

**Perfil:**

Emerson Antonio Klisiewicz.

Possui graduação em Ciências da Computação, pela Pontifícia Universidade Católica do Paraná (1994), Especialização em Sistemas de Informações Gerenciais , pela Pontifícia Universidade Católica do Paraná (1995), Especialização em Redes e Sistemas Distribuídos pela Pontifícia Universidade Católica do Paraná (1997), com experiência de mais de 20 anos em áreas de tecnologia de empresas estatais e financeiras.

Currículo Lattes:

<http://buscatextual.cnpq.br/buscatextual/visualizacv.do?id=K4282839T3>

## **AULA 04 – Requisitos**

**Introdução:**

Nessa quarta aula, estaremos estudando sobre os Requisitos. O tratamento da informação é um requisito que fundamenta o processo de desenvolvimento de software antes da solução de tecnologia a ser aplicada. Para isso é necessário gerenciá-lo de forma eficiente.

**Contextualizando:**

Cada projeto deve ter suas fases de desenvolvimento adequadas às necessidades de tratamento da informação.

**Requisitos**

São descrições dos principais recursos de um produto de software, seu fluxo de informações, comportamento e atributos. Fornece uma estrutura básica para o desenvolvimento de um produto de software. O grau de compreensibilidade, precisão e rigor da descrição fornecida por um documento de requisitos de software tende a ser diretamente proporcional ao grau de qualidade do produto resultante.

**Gerenciamento de Requisitos**

Estabelecer quais funções são requeridas pelo sistema e as restrições sobre a operação e o desenvolvimento do sistema. A Engenharia de Requisitos objetiva fornecer métodos para compreender a natureza de um problema, estabelecer com exatidão o que um sistema deve fazer.

Após a obtenção dos requisitos, podemos/necessitamos realizar um estudo de viabilidade, mostrando as opções ao usuário. O Estudo de Viabilidade que indica se o esforço em desenvolver a idéia vale a pena. Visa tanto a tomada de decisão, como a sugestão de possíveis alternativas de solução. O estudo de viabilidade deve

oferecer informações para ajudar na decisão se o projeto pode ou não ser feito, se o produto final irá ou não beneficiar os usuários interessados. Através dele há a escolha das alternativas entre as possíveis soluções.

Requisitos são inevitavelmente incompletos e inconsistentes e requisitos novos surgem durante o processo de acordo com mudanças nas necessidades do negócio e um entendimento melhor do sistema é desenvolvido. Diferentes pontos de vista têm diferentes requisitos e esses geralmente são contraditórios.

Para implementar uma gerência de requisitos eficaz é necessário, de antemão, definir um conjunto de políticas. É necessário definir um conjunto de objetivos para o processo de gerência. Estes objetivos devem ser claros e transmitidos para todos os integrantes da equipe. Todos os artefatos (documentos) produzidos durante o desenvolvimento do software devem tornar a gerência dos requisitos visível e transparente.

Os aspectos fundamentais de gerência de requisitos. São eles: controle de mudanças, gerência da configuração, rastreabilidade e garantia da qualidade.

#### **a-) Rastreamento**

Responsável por dependências entre requisitos, suas origens e projeto do sistema.

##### **Tipos:**

- Rastreamento de origem: ou seja a associação entre requisitos e stakeholders que propuseram tais requisitos.
- Associação entre requisitos dependentes: rastreamento de projeto.
- Associação dos requisitos com o projeto: rastreamento de requisitos.

#### **b-) Controle de Mudanças**

Em um ambiente real de desenvolvimento de software mudanças são inevitáveis. Em muitos dos casos os requisitos do sistema mudam enquanto o sistema ainda está sendo desenvolvido. As razões para mudanças são de vários tipos, entre outras:

- A complexidade dos sistemas impõe mudanças à medida que se adquire maior conhecimento sobre os mesmos.
- Requisitos errados ou mal definidos precisam ser corrigidos/ajustados ao longo do processo de desenvolvimento.
- Mudanças no ambiente: regras de negócios, leis, políticas internas.
- Desenvolvedores querem adicionar funcionalidades mais avançadas de modo a oferecer vantagem.
- Tecnologia muda.

- Clientes mudam de idéia.

As alterações que precisam ser feitas nos requisitos e artefatos de software. É fundamental que as alterações dos requisitos sejam:

- Identificadas e avaliadas.
- Avaliadas sob o ponto de vista de risco.
- Documentadas.
- Planejadas.
- Comunicadas aos grupos e indivíduos envolvidos.
- Acompanhadas até a finalização.

### c-) Gerência da Configuração

A Gerência de Configuração está comumente associada a dois tipos de tarefas: controle de versões e controle de configuração. Por controle de versões entende-se as atividades associadas a manter, sob estrito acompanhamento, as diferentes versões de um artefato. Por controle de configuração entende-se as atividades associadas a manter, sob estrito acompanhamento, o conjunto de artefatos relacionadas a uma determinada configuração do sistema.

### d-) Garantia de Qualidade

O objetivo da gerência de qualidade de requisitos é garantir que uma base de requisitos composta essencialmente de bons requisitos. Existe uma vasta literatura sobre o que torna um requisito bom, que pode ser resumida através dos seguintes critérios [Young01, Wiegers99]:

Necessidade	O sistema é capaz de atingir seus objetivos sem este requisito? Caso afirmativo este é um requisito desnecessário
Verificável	É possível verificar que este requisito está sendo atendido pelo sistema?
Atingível	O requisito pode ser atendido pelo sistema que está sendo desenvolvido?
Livre de Ambiguidades	O requisito possui mais de uma interpretação possível?
Completo	O documento de especificação do sistema contém todos os requisitos?
Consistente	Todos os requisitos podem ser atendidos sem que se entre em conflito uns com os outros?
Rastreável	A origem dos requisitos é conhecida? O requisito pode ser referenciado e localizado no sistema?
Alocação	O requisito pode ser alocado a um elemento ou componente do sistema?
Concisão	O requisito está descrito de forma simples e concisa?
Livre de Implementação	O requisito descreve o QUE deve ser feito sem influências de possíveis implementações?
Identificador único	Cada requisito possui um identificador único que permita com que possamos referenciá-lo de forma única e precisa?
Correção	O requisito contém toda a informação necessária que permita sua implementação?
Priorizável	O requisito é passível de ser priorizado frente aos outros requisitos?

## **Refinamentos de Requisitos**

Segundo o CMM, uma das atividades da área de processo chave, gerência de requisitos é a revisão dos requisitos antes de incorporá-los ao projeto. É necessário:

- Identificar requisitos incompletos ou ausentes.
- Determinar se os requisitos estão claros, possíveis de serem implementados, consistentes e verificáveis.
- Revisar requisitos com problemas potenciais.
- Negociar compromissos com os grupos envolvidos.

A maior parte dos requisitos de software para sistemas de informação são escritos utilizando-se linguagem natural. Esta falta de formalidade na captura dos requisitos implica em uma série de potenciais problemas. Os problemas mais comuns são:

### **a-) Ambiguidade**

Falta de clareza ou duplo sentido de frases ou expressões. Este tipo de requisito leva a interpretações erradas ou inconsistentes das necessidades reais dos usuários.

### **b-) Requisitos Incompletos**

Requisitos que deixam de fora parte da informação necessária à sua compreensão.

### **c-) Requisitos Múltiplos**

Estes são requisitos que concatenam vários requisitos em um só. Estes requisitos devem ser separados para facilitar a tarefa de priorização e gerência de mudanças.

### **d-) Requisitos com alternativas**

São aqueles requisitos que não estabelecem claramente qual deve ser a ação do sistema frente a uma dada situação. De modo geral contém frases do tipo mas, com exceção, apesar, e quando.

## **Validação de Requisitos**

Nesta fase pretende-se demonstrar que o documento de requisitos produzido corresponde, de fato, ao sistema que o cliente pretende. À semelhança do que sucede na análise dos requisitos, pretende-se encontrar problemas/conflitos na especificação, porém ao contrário das fases anteriores esta fase lida com uma especificação completa dos requisitos. A validação é especialmente importante em sistemas de grandes dimensões uma vez que erros encontrados demasiado tarde (durante o desenvolvimento ou já depois de o sistema estar a ser usado) no documento de requisitos têm repercussões proporcionais à dimensão do projeto. Uma vez que alterações em requisitos já consolidados têm um custo muito superior a alterações no código ou design, este tipo de erro traduz-se em elevados custos e necessidade de refazer muito do trabalho que se julgava já concluído.

Revisões dos requisitos (Verificação de Validade), Prototipação, Geração de casos de teste, Análise de consistência automática, são formas de se validar os requisitos, sempre se buscando a sua completude para a produção de softwares de qualidade.

A fase de validação não deve ser encarada de ânimo leve: trata-se de uma fase muito importante na engenharia de requisitos e da qual dependem elevados custos a médio e longo prazo. Por depender bastante dos retornos transmitidos pelos clientes (que não são peritos em validação de requisitos) é bastante falível e regra geral há erros que não são encontrados num primeiro momento, o que leva inevitavelmente a discordâncias numa fase posterior, já com o documento validado e o projeto em desenvolvimento ou concluído. Quando isto sucede, torna-se necessário negociar e chegar a um acordo quanto à forma de corrigir o erro detectado.

### **Matriz de Rastreabilidade**

Um bom levantamento de requisitos começa sempre pela seleção das melhores fontes de informação que serão usadas para montar a matriz de requisitos, que será matéria-prima para definir o escopo do projeto.

- Regra de Negócio para Caso de Uso
- Termo para Regra de Negócio
- Requisito de Negócio para Caso de Uso
- Requisito de Negócio para Especificação Suplementar

Projeto <nome_projeto> - Matriz de Rastreabilidade				
Requisito	Documento fonte	Arquitetura	Componente	Caso de teste

### **Síntese**

É necessário que o analista saiba se comunicar com os clientes e garantir clareza de ideias colocadas por eles. Programa de computador, documentação, arquivos de configuração, entre outros e existe por causa das necessidades de clientes. Como transformar necessidades em software? Devem ser consideradas as atividades de como entender as necessidades do cliente, planejar a solução, implementar a solução, validar esta solução, entregar o produto ao cliente. Estas atividades são executadas ordenadas ou não, formalmente ou informalmente. Todo

processo de transformação tem início e fim. Essa variável temporal, denominada de ciclo de vida, determina as fases do desenvolvimento de software. Os clientes estarão satisfeitos quando você atender às expectativas, entregar no prazo e entregar tudo dentro do orçamento. O Sucesso começa com a Gerência de Requisitos.

Então nós faremos a Análise do Problema, entenderemos o problema, obteremos a concordância dos envolvidos, e faremos o levantamento dos Requisitos.

No levantamento de requisitos teremos que identificar quem usará o sistema (atores). Temos que descobrir como o sistema será usado (casos de uso).

Feito isso entramos na Gerência de Requisitos onde especificaremos os requisitos completamente, iremos gerenciar expectativas, mudanças e erros e faremos o controle de possíveis aumentos do escopo. Tudo isso sempre com uma equipe definida e informada.