**Unidade 3**

Claudia Werlich

**Engenharia de REQUISITOS**

**Atenção o Requisito é extremamente importante uma falha no ou um esquecimento de algum requisito pode acarretar o fracasso do Software.**

**Ser um profissional de TI é propor soluções é saber o que o cliente precisa e as vezes nem sabe que precisa;**

**O desafio é aplicar e identificar o processo de engenharia de requisitos assim como suas técnicas e análises**

Você será designado a elaborar uma apresentação com os seguintes questionamentos:

O que precisará ser feito?

Quais requisitos funcionais podem ser apontados nesta primeira etapa?

Determine também requisitos não funcionais que contribuem para atender às expectativas do cliente, classificando-os em “requisitos funcionais” e “não funcionais”.

Maiores problemas no desenvolvimento de um software :

É estabe­lecer quanto tempo ele levará para ser concluído.

É “garantir que o software desenvolvido para o cliente seja exatamente como ele deseja”

Tais desafios podem ser ultrapassados conhecendo a engenharia de requisitos;

Será elaborado um documento sobre o conjunto de atividades com a principal meta é definir o que deve ser implementado antes de começar a desenvolver o sistema;

É o processo de descrever todas as funcionalidades que um sistema deve possuir, bem como descrever todos os serviços e as restrições de seu funcionamento;

É proporcionar a todos os envolvidos no desenvolvimento do sistema uma mesma compreensão por escrito do problema;

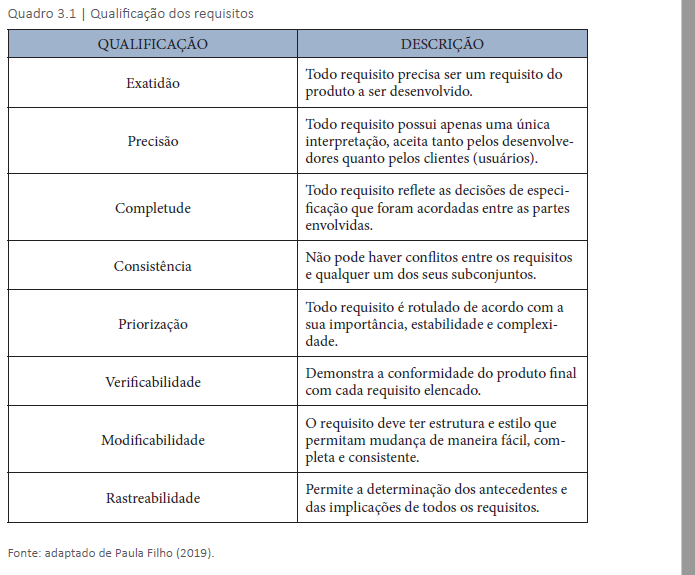
Mas, o que é o requisito de um sistema?

Há algum tempo, requisito de sistema era definido como uma **função** que o software deveria ter ou uma **qualidade** que ele deveria apresentar.

Atualmente, além da **função** e da **qualidade**, os requisitos de um sistema abrangem **especifica­ções dos serviços** que ele deve fornecer, **restrições que deve possuir**, caracte­rísticas gerais e restrições que devem ser atendidas no seu processo de desen­volvimento;

Os requisitos podem evoluir e podem ser modificados no decorrer do desenvolvimento do software, eles não são estáticos e sofrem atuali­zações constantes; além disso, devem ser documentados para fins de controle.

os requisitos demonstrados estão escritos do que o software deverá realmente fazer. Eles devem ser objetivos e consistentes,



Os requisitos são classificados como:

Essencial: Extrema importância (é fundamental) para o software ser executado: Sem esse fica incompleto;

Importante:

é aquele que não é essencial para que o software seja implantado, ou seja, ele deve ser realizado, mas pode ficar em segundo plano; são requisitos desejáveis, porém não imprescindí­veis.

Desejável:

aquele que não é impres­cindível para o software estar concluído, é algo opcional que pode ou não ser realizado e até pode ser eliminado no decorrer do desenvolvimento do sistema.

O escopo do projeto descreve de forma clara o que deverá estar incluso no projeto estabelece os limites das operações que serão reali­zadas juntamente com os recursos a serem utilizados

Definir o escopo do projeto é primordial para determinar seu orçamento e seu cronograma.

Os requisitos de um sistema são as especificações das propriedades que deve ter, das características do que deve fazer, de como deve se comportar e das suas restrições.

Os requisitos descrevem o que o sistema deverá fazer e o que ele não deverá fazer, sem mencionar o

modo como será feito;

Na engenharia de requisitos é importante fazer uma distinção dos diferentes tipos de descrições dos requisitos conforme afirma Sommerville (2011). Eles podem ser classificados como:

**Requisitos de usuário:** descrevem os requisitos funcionais e os não funcionais do sistema ;

**Requisitos de sistema: E**specificam detalhes do sistema ; Resultando em um documento que pode servir como parte do contrato entre os envolvidos no desenvolvimento do sistema ;

REQUISITOS X REGRAS DE NEGÓCIO A

**o que o sistema realizará X como o sistema realizará;**

ações claras e objetivas X algum tipo de condição expressa por termos como: se, quando, somente se

pedido ou uma necessidade X é obrigatório testar, sempre que, etc.

Segundo Sommerville (2011) e Pressman (2016), os requisitos de um sistema podem ser classificados como **requisitos funcionais**, **requisitos não funcionais** e **requisitos de domínio**:

1. Requisitos funcionais: determinam de forma clara e precisa as funcio­nalidades específicas do que o sistema deve ou não realizar.
2. Os **requisitos funcionais** determinam os objetivos específicos do sistema, ou seja, o que ele deve possuir ao final de seu desenvolvimento, segundo afirma Sommerville
3. Requisitos não funcionais: estabelecem restrições sobre as funcio­nalidades do sistema, por meio do estabelecimento de padrões específicos de desenvolvimento, plataforma, tempos de resposta e restrições de acessos; estão relacionados às qualidades que o sistema deverá apresentar.
4. Uma recomendação importante para saber se a especificação do requisito é um requisito não funcional é observar se o assunto tratado pode ser mensu­rado (velocidade, tempo, linguagens,

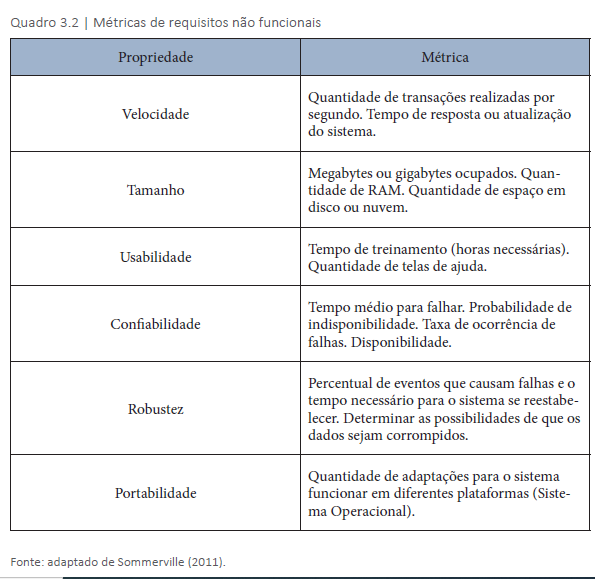
versões); caso seja, possivelmente será um requisito não funcional, conforme Paula Filho

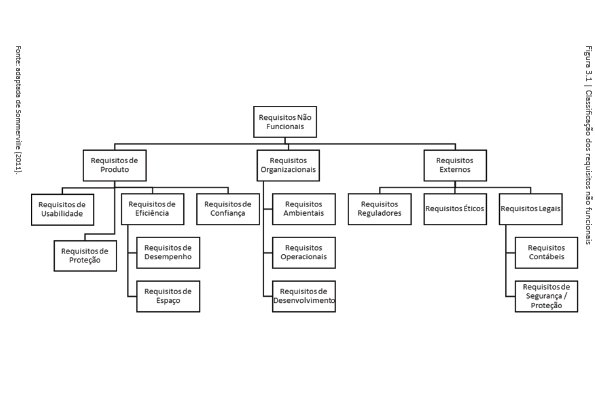
Os requisitos não funcionais possuem uma classificação gerada a partir de três grandes áreas de requisitos:

1. Requisitos de produto: especificam ou restringem o comportamento do sistema e podem determinar a linguagem de programação a ser utilizada.

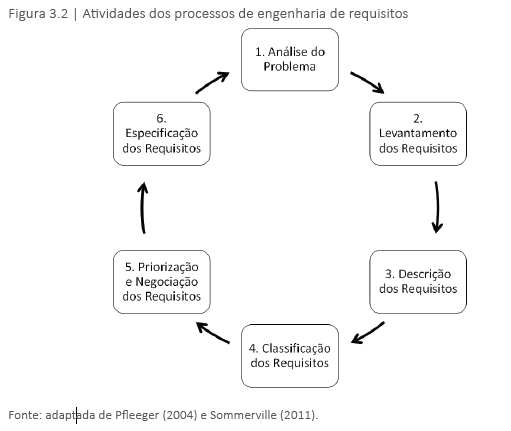
2. Requisitos organizacionais: são derivados das políticas e procedi­mentos da empresa, do cliente e do desenvolvedor.

3. Requisitos externos: abrangem todos os fatores externos ao sistema e que podem influenciar no desenvolvimento do sistema sem desobedecer a legislação vigente.

1. ****



1. Requisitos de domínio: determinam as características do domínio do sistema, refletindo as características do sistema; podem estabelecer restrições aos requisitos funcionais ou podem indicar cálculos especí­ficos sobre determinado requisito.
2. os requisitos de domínio descrevem as características e estabelecem ressalvas aos requisitos funcio­nais, indicando, por exemplo, um cálculo obrigatório para o requisito ser validado.
3. Eles podem ser novos requisitos funcionais ou alguma restrição (complementar) de algum requisito funcional.
4. Um exemplo de requisito de domínio seria a seguinte situação: o aluno será considerado aprovado se obtiver, no mínimo, 3000 pontos nas avaliações semestrais e 4000 pontos nas atividades complementares de cada disciplina. Observe o que está sendo determinado: pontuação mínima das avaliações deve ser igual a 3000 pontos E pontuação mínima deve ser igual a 4000 pontos. As duas condições deverão ser verdadeiras para que o aluno esteja aprovado no sistema.

****

Todas as atividades que englobam a engenharia de requisitos possuem como finalidade a produção de um documento de requisitos, como afirmam Sommerville (2011) e Pressman (2016), englobando as seguintes etapas:

1. **Concepção:** determina-se o escopo geral do sistema e todos os envolvidos.
2. **Elicitação:** faz-se o levantamento dos requisitos funcionais e não funcionais, utilizando técnicas como entrevistas, observação e pesquisa. Extrair o Problema
3. **Elaboração:** detalha-se cada requisito (levantado e escrito em linguagem natural) para transformá-los em modelos conceituais (UML – Linguagem Unificada de Modelagem), eliminando erros, esquecimentos, duplicidades e inconsistências.
4. **Negociação:** identificam-se os requisitos conflitantes, sendo reali­zada uma negociação entre as partes envolvidas para modificar e/ou eliminar o requisito.
5. **Especificação:** transformam-se os requisitos ao se observar a visão do sistema (do desenvolvimento da solução).
6. **Validação:** realiza-se a homologação dos requisitos pelos usuários envolvidos, verificando se todos os requisitos foram atendidos (na visão do cliente).
7. **Gerenciamento:** é a verificação constante (no decorrer dos processos) de que os requisitos estão de acordo com o escopo do projeto e a garantia da rastreabilidade, que é o gerenciamento das eventuais modificações que um requisito pode sofrer.

Resumidamente, os requisitos funcionais explicitam as funções, os comportamentos e as propriedades de entrada e saída de um sistema, e os requisitos não funcionais referem-se aos atributos (qualidades) do sistema.

Requisito funcional não é uma funcionalidade, é uma necessidade funcional (uma função) que o software deve atender.

Atributos requisitos funcional:

**Unidade:** O RF deve propor uma única coisa apenas.

**Completude:** O RF deve ser autocontido, deve ter “início/meio/fim”, ser completo.

**Consistência:** O RF não deve contradizer outro RF do mesmo escopo do projeto.

**Atomicidade:** Um RF para ser atômico precisa também ter unidade, pois atomicidade remete a assumir apenas uma responsabilidade.

**Não-Ambiguidade:** Um RF não pode ser ambíguo, não pode propor algo que não fica claro o que é.

**Verificável:** Não adianta ter um RF se ele não é palpável, possível de associar com um artefato de construção, de testes. Um RF tem que ser testável.

**Rastreável:** Deve ser possível achar o RF no sistema pronto, funcional e executável.

[**Prioridade**](https://www.ateomomento.com.br/priorizacao-de-requisitos/)**:** Um RF Essencial é algo muito diferente de um RF Desejável, possuem valores para o negócio completamente diferentes.

Um detalhe fundamental é o uso do tempo verbal no nome do RF. Um RF, em tempo de especificação, **refere-se a algo que será feito, uma ação a ser realizada** **pelo sistema**. Por isso o nome precisa estar no tempo verbal infinitivo

Requisitos Não Funcionais são premissas ou restrições que o sistema deverá atender, mas que não são realizados através de funcionalidades

 Como o próprio nome diz, é uma “não funcionalidade”, ou seja, trata-se de algo que não é uma funcionalidade, mas que precisa ser realizado para que o software atenda seu propósito.

Toda necessidade que **for realizada através de funcionalidades** é resultado de um ou mais requisitos funcionais (pois uma funcionalidade pode realizar vários requisitos funcionais, não necessariamente apenas um) e toda necessidade **que não puder ser atendida desta forma**, é um Requisito Não-Funcional – geralmente trata-se de premissas e restrições técnicas aplicadas ao projeto. Acho que essa definição, aparentemente simples, define bem um RNF.