

# **Análise de Projetos**

# **Requisito de Software**

**Professora Sabrina Silveira**  
**Colégio Protásio Alves**

A elaboração de um projeto, seja ele para criar um produto novo ou melhorá-lo, surge tipicamente em função de algum problema, oportunidade ou necessidade de negócio.

**A partir do momento em que for decidido iniciar um processo de construção de um software, deve-se definir o escopo do projeto através de uma lista de funcionalidades que se deseja disponibilizar para seus usuários no sistema, no qual estas necessidades identificadas são denominadas de requisito.**

**Portanto, requisito é uma definição formal e detalhada de uma função do sistema.**

Podemos definir **requisitos** como um conjunto de condições ou capacidades necessárias que um software deve possuir para que o usuário possa resolver um problema ou atingir um objetivo, ou para atender as necessidades ou restrições da organização ou dos outros componentes do sistema.

A análise e especificação dos requisitos têm vital importância no desenvolvimento de softwares, pois é nesta etapa da engenharia de software que são levantadas as informações de grande importância para a construção adequada do sistema (considerado marco decisivo de sucesso ou fracasso dos projetos).

**Embora seja óbvio, é importante que se chame a atenção para o fato de que os requisitos são a base sobre a qual serão apoiadas todas as demais atividades do processo de software.**

**Se os requisitos não forem devidamente especificados, o sucesso do projeto poderá estar comprometido desde seu início.**

Embora seja óbvio, é importante que se chame a atenção para o fato de que os requisitos são a base sobre a qual serão apoiadas todas as demais atividades do processo de software. Se os requisitos não forem devidamente especificados, o sucesso do projeto poderá estar comprometido desde seu início.

Problema	%
Requisitos incompletos	13.1
Baixo envolvimento do cliente	10.6
Falta de recursos	12.4
Expectativa não realista	9.9
Falta de suporte gerencial	9.3
Mudanças nos requisitos	8.7
Falta de planejamento	8.1
Requisitos desnecessários	7.5

Estudos mostram porque trabalhar com requisitos de software é tão importante. De acordo com um estudo do Standish Group, no ano de 2014, cinco dos oito principais fatores de falhas em projetos estão relacionados a requisitos. Estes são: requisitos incompletos, baixo envolvimento do cliente, expectativas não realistas, mudanças nos requisitos e requisitos desnecessários.

Como o requisito é uma documentação que diz o que o software deverá realizar, o sucesso do projeto depende de sua definição clara, e a engenharia de requisitos é considerada uma etapa do processo de software muito importante, pois é extremamente árduo quando se elabora requisitos de software de forma vaga e incompleta, gerando retrabalho para a equipe do projeto.

Algumas dificuldades no processo de obtenção de requisitos:

- falta de conhecimento do usuário das suas reais necessidades e do que o produto de software pode lhe oferecer;
- falta de conhecimento do desenvolvedor do domínio do problema;
- domínio do processo de extração de requisitos pelos desenvolvedores de software;
- comunicação inadequada entre desenvolvedores e usuários;
- dificuldade do usuário em tomar decisões;
- problemas de comportamento;
- questões técnicas.

A imagem a seguir demonstra de uma maneira divertida como essa falta de entendimento entre as partes gera confusão e como o resultado pode deixar o cliente insatisfeito.



Diversas atividades de requisitos de software ocorrem ao longo de todo o ciclo de vida do software, um trabalho que consiste na análise de requisitos para identificar, quantificar, definir, especificar, documentar, rastrear, priorizar e classificar os principais problemas que o futuro software deve resolver. A seguir serão apresentados os tipos de requisitos de software.

## Requisitos Funcionais

Os requisitos funcionais devem descrever as funções e as ações do sistema.

Colocando de uma maneira mais simples, **esses requisitos devem dizer o que o sistema faz, ou não faz, em algumas situações**. Lembrando que essa especificação deve ser completa e consistente. Os Requisitos funcionais são aqueles que definem as funções ou ações fornecidas pelo sistema e que o usuário pode utilizar, além das regras de negócio e as interfaces internas e externas.

Os Requisitos funcionais do sistema, variam de requisitos gerais, que indicam o que o sistema deve fazer, até requisitos bem específicos, que refletem os sistemas e as formas de trabalho de uma organização.



## Exemplos de Requisitos funcionais:

- O sistema deve permitir a inclusão, alteração e remoção de professores com os seguintes atributos: nome, endereço, cidade, etc.
- O sistema fornecerá telas apropriadas para o usuário ler documentos.
- O sistema deve gerar diariamente, para cada unidade de ensino, a lista dos alunos inadimplentes.
- O sistema deve permitir pesquisar a lista dos alunos com nota abaixo da média.

Em princípio, a especificação dos requisitos funcionais de um sistema deve ser completa e consistente. Completude significa que todos os serviços requeridos pelo usuário devem ser definidos. Consistência significa que os requisitos não devem ter condições contraditórias. Na prática, para sistemas grandes e complexos, é praticamente impossível alcançar a completude e consistência de requisitos.

## Requisitos não funcionais

Os Requisitos não funcionais, ao contrário dos funcionais, não se relacionam com a funcionalidade do software, mas eles **definem outras propriedades que também são importantes para o sistema, como qualidade, confiabilidade, desempenho, usabilidade, segurança, portabilidade e Integridade.**

Os requisitos não funcionais do sistema, como o nome sugere, são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. **Esses requisitos podem estar relacionados às propriedades como confiabilidade e tempo de resposta.**

Podemos ver que os requisitos não funcionais podem ser provenientes das características requeridas para o software.

Classificação dos requisitos não funcionais:

- Requisitos de produto: Esses requisitos especificam ou restringem o comportamento do software.
- Requisitos organizacionais: Esses são os requisitos gerais de sistemas derivados das políticas e procedimentos da organização do cliente e do desenvolvedor.
- Requisitos externos: Esse tipo abrange todos os requisitos que derivam de fatores externos ao sistema e seu processo de desenvolvimento. Podem incluir requisitos reguladores, que definem o que deve ser feito para que o sistema seja aprovado para uso, tal como um banco central; requisitos legais, que devem ser seguidos para garantir que o sistema opere dentro da lei.

## PROBLEMAS ENCONTRADOS REFERENTES A REQUISITOS

Durante a fase de levantamento de requisitos é necessário que haja uma sintonia entre os analistas, para que o sistema atinja os objetivos. Caso haja problemas nessa fase, o prejuízo pode ser imenso e até mesmo comprometer e inviabilizar todo o sistema. A tirinha apresentada relata bem esse problema.



Muitos projetos têm falhado devido a problemas no levantamento de requisitos, tais como **requisitos incompletos, mal-entendidos ou ambíguos**. Faz-se necessário então que esses requisitos sejam **bem construídos**, de modo que expressem exatamente o que o usuário deseja, da maneira que ele deseja e a forma como poderá ser feito. Por isso é importante conhecer técnicas para tornar a construção destes requisitos mais eficaz e evitar o retrabalho. Existem vários motivos que podem gerar problemas e até mesmo o fracasso de um projeto. A falta de especificação da real necessidade e expectativas dos usuários é o maior motivo dessas falhas, seguido por requisitos incompletos, com baixa qualidade e falta de controle de mudanças.

Por mais que possa parecer estranho, uma dificuldade que podemos encontrar em relação a requisitos é manter o foco no produto e nos objetivos relacionados ao projeto. Devemos atentar se os requisitos apresentados estão compatíveis e alinhados com o projeto. Quando um processo de levantamento de requisitos é ineficaz e não reflete as necessidades reais do cliente, acaba gerando um software de baixa qualidade. Estudos mostram que de 40% a 60% de todos os problemas encontrados em um projeto de software são causados por falhas no processo de requisitos. Se a fase de engenharia de requisitos for bem feita, esses problemas poderiam ser detectados e corrigidos a um custo muito mais baixo.