1 Requisitos

Os requisitos de um sistema/produto de software dizem respeito as necessidades do cliente e como essas precisam ser atendidas dentro do produto final a ser desenvolvido. Assim como Somerville (2011) cita que “os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento.”

O termo requisito pode ser definido em mais de um tipo, devido a grande abrangência do mesmo. Um requisito pode ser detalhado de forma mais ou menos profunda, causando assim, dúvidas com relação a sua especificação correta. Segundo Wiegers (1999) *“no clear, unambiguous undestarding of term 'requirements' exists”*. Em tradução livre, não existe uma declaração clara ou um entendimento não ambíguo do termo 'requisitos'. Com o uso dos tipos corretos é possível analisar a profundidade com maior facilidade.

O requisito deve ser abstraído de forma que represente a necessidade sem a solução ser predefinida em um primeiro caso. Em um próximo momento o requisito seŕa melhor detalhado em forma de descrição de uma funcionalidade, sendo usado pelo desenvolvedor para um entendimento do que deve ser desenvolvido e para que alguns dos *stakeholders* envolvidos com regras de funcionamento do software, possam ter um entendimento completo da solução que será implementada.

1.1 Requisitos de usuário

Os requisitos de usuários são mais gerais, usam de linguagem mais direta, não importando detalhes ou processos de implementação. Segundo Wiegers (1999) “*user requirements describe tasks the users must be able to accomplish with the product.*” Wiegers cita que os usuários deverão ter disponíveis algumas tarefas dentro do sistema, e que essas tarefas descrevem os requisitos de usuário.

Essa apresentação dos requisitos faz com que a leitura para determinados leitores do projeto de software (pessoas que terão acesso ao documento de requisitos) seja facilitada, apresentando apenas pontos importantes e relevantes para estes. Os leitores dos requisitos de usuário são pessoas não interessadas em como o produto será desenvolvido, mas sim o que ele irá cumprir.

Somerville (2011) afirma que “... são declarações em uma linguagem natural com diagramas, de quais serviços o sistema deverá fornecer a seus usuários e as restrições com quais este deverá operar.” O uso de diagramas facilita a visualização e até mesmo a escrita para pessoas mesmo leigas, fazendo com que o cliente consiga demonstrar os requisitos.

1.3 Requisitos de sistema

Os requisitos de sistema utilizam uma linguagem mais detalhada. Este poderá ser utilizado por leitores mais interessados na forma detalhada do desenvolvimento do software. Eles representaram os processos e regras de negócio para cada funcionalidade. Segundo Somerville (2011) os requisitos de sistema “... são descrições mais detalhadas das funções, srviços e restrições operacionais dos sistemas de software.”

O nível de detalhamento nesse formato de requisito irá facilitar a observação de leitores que precisam entender o funcionamento do sistema de forma mais intensa, atingindo assim os desenvolvedores que através dos requisitos de sistema entenderão aquilo que deverá ser implementado.

1.4 Requisitos de negócios

Wiegers (1999) cita ainda requisitos de negócio, uma tradução livre para *bussines requirements*. Estes ficariam acima dos requisitos de usuário, gerando um primeiro contato do cliente com a organização que prestará o serviço de desenvolvimento.

Os requisitos de negócio utilizam objetivos de alto nível da organização que está solicitando o produto de software. Normalmente a utilização dos requisitos de negócio geram documentos com o escopo e a visão do produto a ser desenvolvido.

1.5 Requisitos Funcionais e não funcionais

Os requisitos funcionais e não funcionais são assim caracterizados para ajudar na definição do que será acessível ao usuário do sistema como uma ação e aquilo que será desenvolvido para ser executado sem a percepção do usuário, realizando integrações com normas da instituição, normas de padronização e/ou regras da aplicação como disponibilidade, confiança, usabilidade, etc. Essa divisão permite analisar o que será entregue como uma necessidade para o cliente realizar suas tarefas, como cita Wiegers (1999) em tradução livre, “os requisitos funcionais definem funcionalidades do software que o programadores devem desenvolver no produto para permitir aos usuários completar sua tarefas...”.

Os requisitos não funcionais não ficam visíveis para os clientes, não são como uma funcionalidade requerida para uma ação, mas sim como define Somerville (2011) “... propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área.” Muitas vezes um requisito de usuário é que o sistema tenha uma boa usabilidade, essa questão não faz referência a uma página ou ação do sistema, mas sim como um resultado de todo o processo de implementação do sistema.

Os requisitos não funcionais ainda são definidos como requisitos de produto, organizacionais e externos.

Os requisitos de produtos dizem respeito ao comportamento do produto em si, podendo restringir ou especificando o mesmo. Questões como desempenho, taxa de falha, usabilidade, etc.

Requisitos de organizacionais dizem respeito a normas da organização que irá adquirir o mesmo, como forma de login através do cadastro da empresa, tempo de utilização do sistema. Os requisitos organizacionais ainda dizem respeito aos requisitos da organização que irá desenvolver o sistema, por exemplo, a linguagem de programação a ser utilizada é um requisitos não funcional organizacional.

Requisitos não funcionais externos são aqueles definidos por entidades externas ao cliente e a empresa desenvolvedora, como por exemplo, a ISO. Através de normas a ISO pode definir requisitos para qualidade do sistema. Outros exemplos são orgãos regulamentadores como a receita federal. Segundo Somerville (2011) “esse tipo abrange todos os requisitos que derivam de fatores externos ao sistema e seu processo de desenvolvimento”.

SOMERVILLE, Ian. **Engenharia de software.** 9. ed. São Paulo, SP: Pearson Prentice Hall, 2011. 529 p.

WIEGERS, Karl E. **Software Requirements.**Redmond, Washington: Microsoft Press, 1999. 350 p.