

Regras de Verificação e Análise de Requisitos

1. Objetivo

Este documento tem como objetivo definir as características, padrões e regras que deverão ser obedecidas na especificação, análise e verificação dos requisitos do projeto. O documento orienta a equipe sobre a forma de identificar, descrever e organizar os requisitos, garantindo clareza, consistência, rastreabilidade e alinhamento com as necessidades do cliente.

2. Nomenclatura dos Requisitos

Para garantir padronização e rastreabilidade, as seguintes siglas serão utilizadas:

| Sigla | Tipo de Requisito |
|-------|-------------------------|
| EPXX | Épico |
| HUXX | História de Usuário |
| RFXX | Requisito Funcional |
| RNFXX | Requisito Não Funcional |
| CAXX | Critério de Aceitação |
| RNXX | Regra de Negócio |

3. Organização dos Requisitos por Épico

Os requisitos serão organizados em épicos, sendo que cada épico representará uma entidade principal do sistema ou um grande módulo funcional. Essa estrutura permitirá agrupar histórias de usuário, regras de negócio e critérios de aceitação de forma lógica e rastreável.

- Épico (EP):** Entidade ou macro funcionalidade do sistema (ex.: Usuário, Cliente, Venda).
- História de Usuário (HU):** Requisito detalhado que descreve o que o usuário precisa fazer dentro do épico.
- Critérios de Aceitação (CA):** Condições que definem quando a história é considerada completa.

- **Requisitos Funcionais (RF) e Não Funcionais (RNF):** Associados às histórias ou ao épico correspondente.
- **Regras de Negócio (RN):** Restrições ou condições obrigatórias relacionadas ao comportamento do sistema.

Além disso, os requisitos podem ser genéricos para todo o sistema. Assim, para evitar duplicação de texto, esses requisitos serão organizados em um documento geral para o sistema.

4. Características dos Requisitos

De acordo com o livro *Engenharia de Software*, de Rogério Magela, cada requisito deve obedecer às seguintes características:

1. **Clareza:** O texto deve ser compreensível, sem ambiguidades.
 2. **Consistência:** Não deve haver conflito entre os requisitos.
 3. **Completeness:** Todas as informações necessárias devem estar descritas.
 4. **Viabilidade:** O requisito deve ser realizável técnica e economicamente.
 5. **Rastreabilidade:** Deve ser possível acompanhar o requisito desde sua origem até sua implementação.
 6. **Testabilidade:** O requisito deve permitir verificação por meio de teste, inspeção ou revisão.
-

5. Regras de Especificação e Verificação

Regra 1 – Estrutura por Épico:

Todos os requisitos devem estar organizados dentro de um épico correspondente à entidade ou módulo do sistema. As histórias, regras e critérios deverão estar claramente vinculados a esse épico. Com exceção dos requisitos gerais do sistema.

Regra 2 – Padronização dos Identificadores:

Cada requisito deve possuir um identificador único (sigla + número sequencial) e ser descrito de forma objetiva. As numerações devem seguir a sequência dentro de cada tipo de requisito (ex.: HU01, HU02, RF03, etc.).

Regra 3 – Linguagem Clara e Mensurável:

Evitar termos subjetivos como “fácil”, “rápido” ou “intuitivo”. Sempre que possível, incluir métricas ou parâmetros verificáveis.

Regra 4 – Revisão e Validação:

Antes da aprovação, os requisitos devem passar por:

- **Verificação interna:** checagem de completude, consistência e rastreabilidade pela equipe técnica;
- **Validação externa:** confirmação com o cliente de que o requisito reflete corretamente sua necessidade.

Regra 5 – Controle de Versão e Histórico:

Toda modificação no documento de requisitos deve ser registrada em um histórico de versões, com data, autor e breve descrição das alterações.

6. Conclusão

A aplicação dessas regras garante que o processo de especificação de requisitos seja organizado, rastreável e verificável, assegurando a qualidade das informações e a coerência entre os artefatos de análise e desenvolvimento. A estrutura por épicos favorece a modularização, facilita a priorização e proporciona uma visão clara das funcionalidades associadas a cada entidade do sistema.

7. Referências

MAGELA, Rogério. *Engenharia de Software Aplicada: Fundamentos*. Rio de Janeiro: Alta Books, 2006. ISBN 8576081237.