

Etapas da Identificação de Requisitos

Paulo Vinícius Moreira Dutra

April 2025

1 Identificação dos Requisitos

Essa é a primeira etapa do processo, onde são reconhecidas as necessidades dos stakeholders. O foco é entender o contexto do sistema, os problemas a serem resolvidos e os objetivos do software. **Técnicas usadas:** análise de documentos, análise de sistemas existentes, entrevistas, brainstorming, workshops, observação dos processos e fluxo de trabalho. **Principais desafios:** alinhar diferentes expectativas, evitar ambiguidades.

2 Levantamento dos Requisitos

Aqui os requisitos são coletados de forma mais detalhada, utilizando técnicas específicas como questionários, observação direta e prototipação. **Classificação dos requisitos:**

- **Requisitos Funcionais:** descrevem as funcionalidades específicas que o sistema deve oferecer para atender às necessidades dos usuários. Representam as ações que o sistema deve ser capaz de realizar, como cadastro de usuários, geração de relatórios, processamento de pagamentos, entre outros. Esses requisitos definem o "comportamento observável" do sistema e estão diretamente ligados às interações com o usuário ou com outros sistemas.
- **Requisitos Não Funcionais:** definem restrições e características de qualidade que o sistema deve obedecer, como desempenho, segurança, usabilidade, escalabilidade, compatibilidade, entre outros. Esses requisitos não dizem respeito diretamente às funcionalidades, mas sim a **como** o sistema deve executar suas funções, garantindo confiabilidade, eficiência e uma boa experiência ao usuário.
- **Requisitos de Domínio ou Regras de Negócio:** representam conhecimentos ou restrições específicas da área de atuação do sistema, que influenciam seu comportamento. São regras que definem como os processos devem funcionar de acordo com normas, leis, práticas do setor ou políticas da organização.

3 Validação dos Requisitos

Os requisitos levantados são revisados e validados junto aos stakeholders para garantir que estão corretos, completos e consistentes. **Técnicas usadas:** revisão de requisitos, prototipação, modelagem UML, análise de viabilidade. Essa etapa evita retrabalho futuro ao eliminar inconsistências e requisitos mal definidos.

4 Priorização dos Requisitos

Após a validação, os requisitos são organizados de acordo com sua importância e urgência para o projeto. Isso permite que a equipe de desenvolvimento foque nos aspectos essenciais do sistema primeiro. **Critérios de priorização:**

- Impacto no negócio.
- Complexidade de implementação.
- Dependências entre requisitos.
- Necessidade do usuário final.

Diferentes técnicas podem ser utilizadas, como MoSCoW (Must have, Should have, Could have, Won't have) e Análise de Valor.

Nem todos os requisitos podem ser implementados ao mesmo tempo. A priorização pode ser feita usando o método **MoSCoW**:

- **Must Have (Obrigatório):** requisitos essenciais.
- **Should Have (Importante):** desejáveis, mas não essenciais.
- **Could Have (Opcional):** funcionalidades adicionais.
- **Won't Have (Descartado):** não serão implementadas no momento.

5 Validação e Revisão dos Requisitos

Antes da implementação, os requisitos devem ser validados com os stakeholders para garantir que atendem às expectativas. Isso inclui:

- Revisão detalhada dos requisitos.
- Ajustes para correção de inconsistências.
- Aprovação final.

6 Publicação dos Requisitos

Após a priorização, os requisitos são formalmente documentados e disponibilizados para toda a equipe de desenvolvimento. **Documentação pode incluir:**

- **Especificação de Requisitos de Software (ERS).**
- **Casos de Uso e Diagramas UML.**
- **Critérios de Aceitação.**

Isso garante que toda a equipe tenha uma referência única e alinhada para seguir no desenvolvimento.

Além da documentação, é realizada a **criação do Plano de Liberações**, que define as versões incrementais do sistema a serem entregues ao longo do tempo. Cada versão contempla um conjunto de requisitos priorizados, considerando fatores como valor de negócio, complexidade técnica e dependências entre funcionalidades. O plano visa facilitar o acompanhamento do progresso, reduzir riscos e permitir entregas contínuas e evolutivas, com foco na geração de valor desde as primeiras iterações.

A seção 6.1, apresenta um plano de liberações incremental para o sistema fictício de **Gestão de Pedidos**. O objetivo é distribuir os requisitos de forma planejada em versões entregáveis, facilitando a evolução contínua do sistema e a validação constante das funcionalidades.

6.1 Release 1 - Exemplo Plano de Liberações

Iteração 1: Configuração Inicial e Desenvolvimento Básico

- **Data Inicial:** 01/03/2024 **Data Final:** 15/03/2024
- **Objetivo:** Estabelecer as bases do projeto e implementar funcionalidades essenciais para o cadastro de usuários e consulta de restaurantes.
- **Atividades:**
 1. **Configuração do Ambiente:** Configurar IDE, versionamento de código e ambientes de teste.
 2. **Estrutura do Projeto:** Criar a estrutura básica do projeto, incluindo organização de pastas e arquivos.
 3. **Cadastro de Usuários:** Desenvolver telas e lógica para permitir que usuários se cadastrem no sistema.
 4. **Consulta de Restaurantes:** Implementar funcionalidade para os clientes navegarem e visualizarem informações sobre os restaurantes disponíveis.

Iteração 2: Desenvolvimento de Funcionalidades Principais

- **Data Inicial:** 16/03/2024 **Data Final:** 30/03/2024
- **Objetivo:** Implementar as funcionalidades principais relacionadas à realização de pedidos, opções de pagamento e acompanhamento de pedidos via web.
- **Atividades:**
 1. **Realização de Pedidos:** Desenvolver funcionalidades essenciais para os clientes realizarem pedidos online.
 2. **Opções de Pagamento:** Integrar diferentes formas de pagamento, como cartão de crédito e débito.
 3. **Módulo Web de Acompanhamento:** Desenvolver um módulo web para acompanhamento de pedidos em tempo real.

Iteração 3: Refinamento e Melhorias

- **Data Inicial:** 01/04/2024 **Data Final:** 15/04/2024
- **Objetivo:** Refinar funcionalidades existentes com base no feedback dos usuários e adicionar novas funcionalidades, como notificações de estoque e promoções, e funcionalidades de gestão integrada.
- **Atividades:**
 1. **Refinamento com Feedback:** Revisar funcionalidades existentes e fazer ajustes com base no feedback dos usuários.
 2. **Notificações de Estoque e Promoções:** Implementar sistema de notificações para restaurantes sobre estoque baixo e promoções.
 3. **Gestão Integrada:** Adicionar funcionalidades de gestão de estoque, pedidos e usuários para os restaurantes.

Iteração 4: Testes Finais e Preparação para Lançamento

- **Data Inicial:** 16/04/2024 **Data Final:** 30/04/2024
- **Objetivo:** Realizar testes finais, corrigir problemas identificados e preparar o sistema para o lançamento.
- **Atividades:**
 1. **Testes de Integração e Aceitação:** Realizar testes finais de integração e aceitação do sistema.
 2. **Correção de Bugs:** Identificar e corrigir quaisquer problemas ou bugs encontrados durante os testes.
 3. **Preparação para Lançamento:** Finalizar a documentação do usuário, preparar o ambiente de produção e realizar uma revisão final do sistema antes do lançamento.