

Projeto Integrador de Tecnologia da Informação II

Prof. Dr. Awdren de Lima
Fontão

Módulo 1 - Resolução de Problemas e Análise de Requisitos

Unidade 1 - Identificação e definição de problemas

Objetivos

- Compreender a importância da identificação correta de problemas;
- Explorar metodologias e técnicas para definir problemas;
- Relacionar a identificação de problemas à Engenharia de Software e Gestão Ágil.

Por que identificar problemas é essencial?

- A base para qualquer solução eficiente é um problema bem definido;
- Evita desperdício de tempo e recursos;
- Facilita o alinhamento entre *stakeholders* e equipe;
- Impacta a satisfação do usuário e o sucesso do produto.

O que é um problema?

- Um desafio que precisa ser resolvido por meio de *software*.
- Características:
 - Claro e específico;
 - Mensurável;
 - Relevante;
 - Viável.

Etapas de Identificação de problemas

1. Observação e Coleta de Dados;
2. Formulação do Problema;
3. Validação do Problema.

Técnicas para descobrir e definir problemas

1. 5 Porquês;
2. Mapa de Empatia;
3. Análise SWOT;
4. Diagrama de Ishikawa.

5 porquês

- Técnica para identificar a causa raiz de um problema.
- Perguntar "Por quê?" sucessivamente até chegar à origem do problema.
- Exemplo:
 - O sistema está lento. Por quê?
 - Porque há muitas requisições. Por quê?
 - O banco de dados está sobrecarregado. Por quê?
 - Falta otimização na consulta. Por quê?
 - Há queda do sistema. Por quê?

Mapa de empatia

- Técnica para entender as necessidades do usuário.
- Perguntas essenciais:
 - O que ele pensa e sente?
 - O que ele vê?
 - O que ele escuta?
 - O que ele fala e faz?
- Permite desenvolver soluções mais alinhadas com a realidade dos usuários.

Análise SWOT

- Ferramenta de análise estratégica. Examina:
 - *Strengths* (Forças): O que já funciona bem?
 - *Weaknesses* (Fraquezas): O que precisa ser melhorado?
 - *Opportunities* (Oportunidades): O que pode ser explorado?
 - *Threats* (Ameaças): O que pode prejudicar a solução?
- Usado para entender o contexto do problema.

Diagrama de Ishikawa

- Também chamado de diagrama de causa e efeito.
- Organiza possíveis causas de um problema em categorias.
- Exemplo:
 - Problema: Queda na qualidade do *software*.
 - Causas possíveis:
 - Falha humana;
 - Falta de testes;
 - Problema no ambiente de produção.

Algumas boas práticas

- Evitar generalizações;
- Não assumir soluções antes de entender o problema;
- Validar com usuários reais antes de implementar.

Conclusão

- Identificar bem um problema leva a soluções eficazes;
- Técnicas estruturadas ajudam a evitar vieses;
- Validação com usuários e *stakeholders* é essencial.

Referências

VALENTE, Marco Tulio. **Engenharia de software moderna**: princípios e práticas para desenvolvimento de software com produtividade. v. 1, n. 24, 2020. Disponível em: <https://link.ufms.br/ye4Sd>. Acesso em 13 jan. 2025. **Trecho a ser lido: Capítulo 1 - Seção 1.2 - O que se estuda em Engenharia de Software?**

MUNIZ, Antonio *et al.* **Jornada ágil do produto**. 1. ed. Rio de Janeiro: Brasport, 2020. 1 recurso on-line. ISBN: 9786599062193. Disponível na Biblioteca Digital da UFMS. **Trecho a ser lido: Parte VI - Início do produto - Ferramentas para descobrir soluções.**

Licenciamento



Respeitadas as formas de citação formal de autores de acordo com as normas da ABNT NBR 6023 (2018), a não ser que esteja indicado de outra forma, todo material desta apresentação está licenciado sob uma [Licença Creative Commons - Atribuição 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).

