**1. Fundamentos da Engenharia de Requisitos**

**Definição e importância:**

**Requisitos como condição ou capacidade necessária para resolver problemas ou alcançar objetivos (IEEE, 1990).**

**Impacto de requisitos mal compreendidos (exemplos: Aeroporto de Denver, Ariane 5, Boeing 737 Max).**

**Ciclo de vida do software:**

**Concepção, Desenvolvimento, Manutenção, Descontinuação.**

**Atividades em cada fase (especificação, implementação, testes, etc.).**

**2. Stakeholders e Técnicas de Elicitação (Aula 02)**

**Stakeholders:**

**Papéis: Analista de Requisitos, Product Owner (PO), Clientes, Usuários.**

**Regras do projeto:**

**Método ágil SCRUM, Sprints, Sprint Review e Retrospective.**

**Papéis rotativos (PO, QA).**

**Técnicas de elicitação:**

**Brainstorming: geração de ideias sem críticas iniciais.**

**Elevator Pitch: apresentação concisa de ideias.**

**Observação e JAD (Joint Application Design): reunião estruturada com papéis definidos.**

**3. Tipos de Requisitos (Aula 05)**

**Classificação principal:**

**Funcionais: O que o sistema deve fazer (ex: cadastro, login).**

**Não funcionais: Qualidades e restrições (ex: desempenho, segurança).**

**Categorias:**

**Produto: Atributos do software (ex: usabilidade).**

**Projeto: Restrições de desenvolvimento (ex: linguagem PHP).**

**Processo: Métodos e padrões (ex: entregas a cada 15 dias).**

**4. Características de um Bom Requisito**

**Critérios:**

**Não ambíguo, verificável, determinístico, rastreável, correto.**

**Exemplos:**

**Evitar ambiguidades (ex: "boa usabilidade" é subjetivo).**

**Especificar claramente (ex: "tempo de resposta de 30 segundos").**

**5. Etapas da Engenharia de Requisitos**

**Desenvolvimento:**

**Elicitação: Investigação ativa (entrevistas, workshops).**

**Análise: Detalhamento e resolução de conflitos.**

**Especificação: Documentação (textos, diagramas).**

**Validação: Revisão e testes.**

**Gerenciamento:**

**Rastreabilidade e controle de mudanças.**

**Ferramentas: Trello, matrizes de rastreabilidade.**

**6. Ferramentas e Práticas**

**Ferramentas:**

**Gerenciamento: Trello (Sprints, tarefas).**

**Modelagem: Astah, Lucidchart.**

**Práticas ágeis:**

**Product Backlog, Sprint Backlog.**

**MVP (Produto Mínimo Viável) para validação inicial.**

**7. Exemplos e Exercícios**

**Classificação de requisitos:**

**Exemplo: "O sistema deve usar PHP e MySQL" → Requisito de projeto.**

**Refinamento de requisitos:**

**Transformar "boa usabilidade" em critérios mensuráveis (ex: "tempo de aprendizado ≤ 5 minutos").**

**8. Referências e Leituras Recomendadas**

**Livros:**

**"Software Requirements" (Wiegers e Beatty).**

**"Engenharia de Software" (Pressman e Maxim).**

**Artigos/Casos:**

**Falhas históricas (Ariane 5, Boeing 737 Max).**

**Dicas para Estudo**

**Prática: Classificar requisitos de exemplos reais.**

**Revisão: Criar checklist para verificar qualidade dos requisitos.**

**Aplicação: Simular elicitação com técnicas como Brainstorming ou JAD**