5409 – Engenharia de Software

20241011 – Luis Cardoso

Atividade1 - Analise dos temas dados

**Faça uma breve analise dos temas dados na sessão**

### O que é Engenharia de Software?

### Para que serve a Engenharia de Software?

### 1. Ciclo de Vida de Software (SDLC)

### 2. Análise de Requisitos

### 3. Projeto de Software (Design)

### 4. Implementação (Codificação)

### 5. Testes de Software

### 6. Manutenção de Software

### 7. Gerenciamento de Configuração

### 8. Engenharia de Requisitos

### 9. Qualidade de Software

### O que é Engenharia de Software?

Engenharia de Software é o campo responsável pelo desenvolvimento, manutenção e gestão de sistemas de software de maneira sistemática e controlada. Envolve o uso de metodologias, técnicas e ferramentas para garantir que o software seja criado de forma eficiente e confiável. Trata-se de uma disciplina que combina ciência da computação, matemática e práticas de engenharia, com foco na produção de software de qualidade.

### Para que serve a Engenharia de Software?

O principal propósito da Engenharia de Software é garantir a criação de software de alta qualidade, com eficiência e escalabilidade. Ela busca assegurar que os sistemas atendam aos requisitos funcionais e não funcionais de usuários e organizações, controlando custos, prazos e qualidade, enquanto gerencia riscos associados ao desenvolvimento.

### 1. **Ciclo de Vida de Software (SDLC)**

O SDLC (Software Development Life Cycle) é um processo estruturado que define as etapas de desenvolvimento de um software, desde o planejamento inicial até a sua descontinuação. Ele inclui fases como análise de requisitos, design, codificação, testes e manutenção, garantindo uma abordagem sistemática para o desenvolvimento.

### 2. **Análise de Requisitos**

Esse processo identifica e documenta as necessidades e expectativas dos stakeholders. A análise de requisitos distingue entre **requisitos funcionais** (o que o sistema faz) e **não funcionais** (desempenho, segurança, etc.), sendo crucial para criar um sistema que atenda às expectativas.

### 3. **Projeto de Software (Design)**

O design de software envolve a criação de uma solução técnica que atende aos requisitos identificados. Isso inclui definir a **arquitetura** do sistema, os componentes e as interfaces, determinando como diferentes partes do sistema irão interagir entre si e com o usuário.

### 4. **Implementação (Codificação)**

A implementação é a etapa prática de desenvolvimento do software, onde o design é convertido em código. Envolve a programação em uma ou mais linguagens e a integração de módulos para formar um sistema funcional.

### 5. **Testes de Software**

Os testes garantem que o software funcione conforme o esperado e esteja livre de defeitos. Eles incluem testes unitários, de integração, de sistema e de aceitação, verificando que o software atende aos requisitos e funciona corretamente em diferentes cenários.

### 6. **Manutenção de Software**

Manutenção envolve a correção de erros, aprimoramentos e adaptações que surgem após a entrega do software. Ela garante a continuidade do funcionamento do sistema ao longo do tempo, acomodando novos requisitos e ambientes.

### 7. **Gerenciamento de Configuração**

Essa prática controla as mudanças no software e em sua documentação ao longo do ciclo de vida. Ela inclui **controle de versão** e **gerenciamento de mudanças**, garantindo que o software e suas atualizações sejam consistentes e rastreáveis.

### 8. **Engenharia de Requisitos**

A engenharia de requisitos trata da elicitação, análise, documentação e validação dos requisitos. O processo garante que os requisitos sejam compreendidos e gerenciados adequadamente ao longo de todo o ciclo de desenvolvimento.

### 9. **Qualidade de Software**

Refere-se ao grau de conformidade do software com os requisitos especificados. Envolve **confiabilidade**, **eficiência**, **usabilidade** e **manutenibilidade**, focando em assegurar que o software funcione bem e seja fácil de manter e usar.

Essa análise resume os principais aspectos e etapas do desenvolvimento de software, ressaltando a importância de um processo sistemático para criar soluções de qualidade.