

PLANO DA DISCIPLINA - ENGENHARIA DE SOFTWARE

1. IDENTIFICAÇÃO DA DISCIPLINA

- **Nome:** Engenharia de Software
- **Curso:** Informática e Sistemas de Informação
- **Regime:** Semestral
- **Ano/Semestre:** 5º ano, 1º semestre
- **Carga Horária Total:** 80 horas
- **Horas Semanais:** 5 horas (2 teóricas, 2 teórico-práticas, 1 prática)
- **Precedência Obrigatória:** Programação com Objetos

2. OBJETIVOS GERAIS

- Compreender os fundamentos da Engenharia de Software e suas áreas de intervenção.
- Explorar a importância das metodologias de desenvolvimento de software e sua evolução histórica.
- Desenvolver competências em modelagem e gestão de requisitos de sistemas de informação.
- Aplicar práticas modernas, incluindo UML, metodologias ágeis e padrões de desenvolvimento.

3. COMPETÊNCIAS A DESENVOLVER

- Identificação e levantamento de requisitos de software.
- Utilização da linguagem UML para modelagem de software.
- Aplicação de metodologias de desenvolvimento de sistemas de informação.
- Implementação de boas práticas em engenharia de software.

4. CONTEÚDO PROGRAMÁTICO

UNIDADE 1: Introdução à Engenharia de Software (15h)

- Definição e conceitos básicos.
- Áreas de intervenção e evolução histórica.
- Importância da engenharia de software no desenvolvimento de sistemas modernos.

UNIDADE 2: Gestão de Requisitos (20h)

- Conceito e importância dos requisitos.
- Técnicas de elicitação e gestão de requisitos.
- Especificação e documentação de requisitos.

UNIDADE 3: Modelagem com UML (20h)

- Introdução à UML e suas diferentes visões.
- Modelagem de casos de uso, estrutura e comportamento.
- Representação arquitetural e boas práticas de modelagem.

UNIDADE 4: Metodologias de Desenvolvimento de Software (15h)

- Apresentação das principais metodologias.
- Ciclo de vida tradicional e suas fases.
- Introdução à metodologia RUP e sua aplicação.

UNIDADE 5: Novos Paradigmas e Tendências na Engenharia de Software (10h)

- Desenvolvimento baseado em modelos (MDA).
- Padrões de análise e design.
- Metodologias ágeis e integração contínua.

5. METODOLOGIA DE ENSINO

- Aulas expositivas e interativas.
- Desenvolvimento de estudos de caso e projetos práticos.
- Utilização de ferramentas CASE para modelagem.
- Aplicação de metodologias ágeis em projetos reais.

6. CRITÉRIOS DE AVALIAÇÃO

- **Trabalhos práticos e projetos:** 30%
- **Testes intermediários:** 30%
- **Exame final:** 40%
- **Frequência mínima:** 75% nas aulas práticas.

7. BIBLIOGRAFIA

Bibliografia Básica

- Pressman, Roger S. – *Engenharia de Software*, 4ª Edição, McGraw-Hill.
- Pressman, Roger & Ince, Darrel – *Software Engineering: A Practitioner's Approach*, McGraw-Hill.
- Silva, Alberto & Videira, Carlos – *UML, Metodologias e Ferramentas CASE*, Centro Atlântico.

Bibliografia Complementar

- Sommerville, Ian – *Software Engineering*, Addison-Wesley.
- Larman, Craig – *Applying UML and Patterns*, Prentice Hall.
- Fowler, Martin – *UML Distilled*, Addison-Wesley.

