

**DIRETORIA DE ENSINO/ DEPARTAMENTO DE ENSINO**  
**COORDENAÇÃO DO CURSO: TÉCNICO EM INFORMÁTICA**  
**PROGRAMA DE UNIDADE DIDÁTICA – PUD**

<b>DISCIPLINA: ENGENHARIA DE SOFTWARE</b>
<b>Código: TI.035</b>
<b>Carga Horária Total:</b> 80 CH Teórica:30 CH Prática: 50
<b>Número de Créditos:</b> 4
<b>Pré-requisitos:</b> Nenhum
<b>Semestre:</b> 3º
<b>Nível:</b> Técnico
<b>EMENTA</b>
Definição e evolução da Engenharia de Software; Processos e modelos de processos de desenvolvimento de software; Desenvolvimento ágil de software  Engenharia de Requisitos; Tópicos de Projeto de Software; Modelagem de Sistemas com UML; Padrões de projeto; Testes de software.
<b>OBJETIVOS</b>
Compreender, praticar e aplicar técnicas de engenharia de software em sistemas de informação: <ul style="list-style-type: none"><li>• Técnicas de Gerenciamento de Projetos</li><li>• Práticas de levantamento de requisitos, especificação e prototipação;</li><li>• Técnicas de análise e projeto OO, utilizando ferramentas CASE •</li></ul> Técnicas de teste: ferramentas e metodologia.

<b>PROGRAMA</b>
<b>UNIDADE I: GERENCIAMENTO DE PROJETOS</b> <ul style="list-style-type: none"> <li>• Conceitos, terminologia e contexto de gerência de projetos.</li> <li>• Ciclo de vida de produto e projeto.</li> <li>• Processos de gerência de projetos e gerenciamento ágil de projetos.</li> </ul> <b>UNIDADE II: LEVANTAMENTO DE REQUISITOS</b> <ul style="list-style-type: none"> <li>• Definição de requisitos de produto, restrições, fronteiras de um sistema.</li> <li>• Especificação e documentação de requisitos.</li> <li>• Construção de protótipos para validar requisitos.</li> </ul> <b>UNIDADE III: ANÁLISE E PROJETO</b> <ul style="list-style-type: none"> <li>• Teorias, métodos, técnicas e ferramentas associadas ao projeto de software •</li> <li>• Técnicas orientadas a objeto para análise e projeto de sistemas. •</li> <li>• Diagramas UML de Análise e Projeto.</li> </ul> <b>UNIDADE VI: TESTE DE SOFTWARE</b> <ul style="list-style-type: none"> <li>• Testes de unidade, teste funcional (caixa preta).</li> <li>• Testes de integração, sistema e de aceitação.</li> <li>• Ferramentas de teste (combinação com ferramentas de integração contínua).</li> </ul>
<b>METODOLOGIA DE ENSINO</b>
<ul style="list-style-type: none"> <li>• Aulas expositivas dialogadas,</li> <li>• Atividades práticas no laboratório,</li> <li>• Visitas técnicas em empresas e/ou associações.</li> </ul>
<b>RECURSOS</b>
<ul style="list-style-type: none"> <li>• Laboratório de informática.</li> <li>• Projetor e computador</li> <li>• Lousa e pincel</li> <li>• Ferramentas de Engenharia de Software</li> </ul>
<b>AVALIAÇÃO</b>
<p>Avaliação do conteúdo teórico; Avaliação das atividades desenvolvidas em laboratório. Os alunos serão avaliados pela sua participação nas atividades propostas e por meio de provas teóricas, práticas e trabalhos</p>
<b>BIBLIOGRAFIA BÁSICA</b>
<p>SOMMERVILLE, Ian. <b>Engenharia de software</b>. 9. ed. São Paulo, SP: Pearson, 2011.</p> <p>TONSIG, Sérgio Luiz. <b>Engenharia de software: análise e projeto de sistemas</b>. 2. ed. rev. e ampl. Rio de Janeiro: Ciência Moderna, 2013.</p> <p>WAZLAWICK, Raul Sidnei. <b>Análise e projeto de sistemas de informação orientados a objetos</b>. 2. ed. Rio de Janeiro: Elsevier, 2011.</p>

## BIBLIOGRAFIA COMPLEMENTAR

ENGHOLM Jr, Hélio. **Engenharia de software na prática**. São Paulo: Novatec, 2013.

HIRAMA, Kechi. **Engenharia de software**: qualidade e produtividade com tecnologia. Rio de Janeiro: Campusl, 2012.

MCLAUGHLIN, Brett; POLLICE, Gary; WEST, David. **Use a cabeça! análise e projeto orientado ao objeto**. Rio de Janeiro: Starlin Alta Consult, 2007.

PFLEEGER, Shari Lawrence. **Engenharia de Software: teoria e prática**. 2.ed. São Paulo: Prentice Hall, 2004.

VAZQUEZ, Carlos Eduardo, SIMÕES, Guilherme Siqueira. **Engenharia de Requisitos: software orientado ao negócio**. 1.ed. Rio de Janeiro: Brasport, 2016

**Coordenador do Curso**

---

**Setor Pedagógico**

---