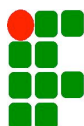


PROGRAMA DE UNIDADE DIDÁTICA – PUD

DISCIPLINA: ENGENHARIA DE SOFTWARE	
Código:	
Carga Horária:	80h
Número de Créditos:	4
Código pré-requisito:	
Semestre:	S5
Nível:	Bacharelado
EMENTA	
Uma Visão Geral sobre Processos. Modelos de Desenvolvimento de Software. Engenharia de Requisitos. Projeto de Interface com o Usuário. Testes de Software. Gerência e Configuração de Mudanças. Gestão de Qualidade.	
OBJETIVO	
Capacitar o aluno a: ▪ Aplicar os processos corretos ao desenvolver um software. ▪ Possuir uma visão geral dos processos de engenharia de software. ▪ Compreender como os processos de desenvolvimento de software estão organizados. ▪ Adotar técnicas que garantam a qualidade do software.	
PROGRAMA	
1. PROCESSOS 1.1. Visão Geral 1.2. Modelo de Processo 1.3. Ferramentas CASE 2. ENGENHARIA DE REQUISITOS 2.1. Conceitos básicos de Levantamento de Requisitos, Análise e Projeto de Sistemas 2.2. Abstração e Concepção dos elementos e das funcionalidades dos sistemas 2.3. Conceito de casos de uso e atores 2.4. Especificação e detalhamento de casos de uso 3. PROJETO DE INTERFACE COM O USUÁRIO 4. TESTES DE SOFTWARE 4.1. Abordagem do teste de software 4.2. Conceitos básicos, tipos de testes e aplicações 4.3. Especificação de teste 4.4. Plano de teste 5. GERÊNCIA DE CONFIGURAÇÃO E MUDANÇA 6. GESTÃO DE QUALIDADE DE SOFTWARE	
METODOLOGIA DE ENSINO	
Aulas expositivas dialogadas, leitura e interpretação de textos, atividades práticas no laboratório, resolução de problemas, leitura e interpretação de textos, seminários.	
AVALIAÇÃO	



A avaliação é um processo contínuo onde serão considerados aspectos qualitativos e quantitativos envolvidos no processo de ensino-aprendizagem no qual os alunos serão avaliados desde a sua participação nas atividades propostas, pontualidade, através de provas teóricas e práticas, participação em sala de aula.

BIBLIOGRAFIA BÁSICA

1. PAULA FILHO, Wilson de Pádua. **Engenharia de Software**. Rio de Janeiro: LTC, 2001.
2. PFLEEGER, Shari Lawrence. **Engenharia de Software: Teoria e Prática**. 2. ed. São Paulo: Prentice-Hall, 2004.
3. PRESSMAN, Roger S. **Engenharia de Software**. São Paulo: Makron Books, 1995.
4. SOMMERVILLE, Ian. **Engenharia de Software**. 6. ed. São Paulo: Addison-Wesley, 2003.

BIBLIOGRAFIA COMPLEMENTAR

1. BASTOS, Anderson. et al. **Base de Conhecimento em Teste de Software**. São Paulo: Martins Editora, 2007.
2. BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML: Guia do Usuário**. 2 ed. Rio de Janeiro: Editora Campus, 2006.
3. FURLAN, J. D.; FELICIANO NETO, A.; HIGA, W. **Engenharia da informação**. São Paulo: McGraw Hill, 1988.
4. GUEDES, G.T.A. **UML: Uma Abordagem Prática**. 2 ed. São Paulo: Novatec, 2008.
5. KRUCHTEN, Philippe. **Introdução ao RUP**. Rio de Janeiro: Ciência Moderna, 2003.
6. LARMAN, C. **Utilizando UML e Padrões**. 3 ed. Porto Alegre: Editora Bookman, 2007.
7. ROCHA, A. R. C. **Qualidade de software: teoria e prática**. São Paulo: Prentice-Hall, 2004.
8. RUMBAUGH, J. **Modelagem e Projetos Baseados em Objetos**. São Paulo: Campus.

Coordenador do Curso

Setor Pedagógico
