



UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CAMPUS ARARANGUÁ-ARA  
CURSO DE TECNOLOGIAS DA INFORMAÇÃO E COMUNICAÇÃO  
PLANO DE ENSINO

SEMESTRE 2012.1

I. IDENTIFICAÇÃO DA DISCIPLINA:

CÓDIGO	NOME DA DISCIPLINA	Nº DE HORAS-AULA SEMANAIS		TOTAL DE HORAS-AULA SEMESTRAIS
		TEÓRICAS	PRÁTICAS	
ARA 7130	ENGENHARIA DE SOFTWARE II	3	1	72

HORÁRIO		MÓDULO
TURMAS TEÓRICAS	TURMAS PRÁTICAS	Presencial
04651 – 2-1010-2 e 3-1010-2	04651 – 2-1010-2 e 3-1010-2	
04652 – 5-2020-2 e 6-2020-2	04652 – 5-2020-2 e 6-2020-2	

II. PROFESSOR(ES) MINISTRANTE(S)

Luciana Bolan Frigo

III. PRÉ-REQUISITO(S)

CÓDIGO	NOME DA DISCIPLINA
ARA 7124	Engenharia de Software I

IV. CURSO(S) PARA O(S) QUAL(IS) A DISCIPLINA É OFERECIDA

Graduação em Tecnologias da Informação e Comunicação (TIC)

V. JUSTIFICATIVA

O profissional responsável por desenvolver sistemas computacionais necessita conhecer e aplicar as principais metodologias adotadas pelo mercado de trabalho para desempenhar sua função com qualidade e ser competitivo no mercado.

VI. EMENTA

...volução da prática de desenvolvimento de software; Critérios de qualidade de artefatos de software; modelos de ciclo de vida; metodologias de desenvolvimento de software; manutenção de software; engenharia reversa; modelagem formal de sistemas; abordagens voltadas ao reuso de software; teste de software; gerenciamento do processo de produção de software e técnicas de apoio ao gerenciamento do processo de produção de software; apoio automatizado ao desenvolvimento de software.

VII. OBJETIVOS

**Objetivo Geral:**

Fornecer subsídios ao aluno para que ele possa compreender os processos de desenvolvimento, implementação e manutenção de software.

**Objetivos Específicos:**

O aluno ao final do curso deve possuir habilidades para:

- Definir engenharia de software explicitando seus conceitos e objetivos;
- Conhecer e aplicar o conceito destinado aos processos de software;
- Conhecer os modelos de ciclo de vida;
- Entender o que é um software de qualidade e conhecer as métricas existentes.

## VIII. CONTEÚDO PROGRAMÁTICO

### Parte I: Fundamentos em Engenharia de Software

- Visão geral da engenharia de *software*
- Desenvolvimento de processo de *software*
- Verificação e Validação de *software*

### Parte II: Gerenciamento

- Gerenciamento da qualidade
- Aprimoramento dos processos

## IX. METODOLOGIA DE ENSINO / DESENVOLVIMENTO DO PROGRAMA

Exposição dialogada, utilizando projetores de slides, trabalhos dirigidos com levantamento bibliográfico e atualização de assuntos, bem como todos os equipamentos necessários para o desenvolvimento das aulas laboratoriais de acordo com cada assunto ministrado; Material de apoio postado no Moodle. Desenvolvimento de trabalhos e exercícios semanais.

## X. METODOLOGIA E INSTRUMENTOS DE AVALIAÇÃO

- A verificação do rendimento escolar compreenderá **frequência e aproveitamento** nos estudos, os quais deverão ser atingidos conjuntamente. Será obrigatória a frequência às atividades correspondentes a cada disciplina, ficando nela reprovado o aluno que não comparecer, no mínimo a 75% das mesmas.
- A nota mínima para aprovação na disciplina será 6,0 (seis). ( Art. 69 e 72 da Res. nº 17/CUn/1997).
- O aluno com frequência suficiente (FS) e média das notas de avaliações do semestre entre 3,0 e 5,5 terá direito a uma nova avaliação no final do semestre (REC), exceto as atividades constantes no art.70,§ 2º. A nota será calculada por meio da média aritmética entre a média das notas das avaliações parciais (MF) e a nota obtida na nova avaliação (REC). (Art. 70 e 71 da Res. nº 17/CUn/1997).

$$NF = \frac{MF + REC}{2}$$

- Ao aluno que não comparecer às avaliações ou não apresentar trabalhos no prazo estabelecido será atribuída nota 0 (zero). (Art. 70, § 4º da Res. nº 17/CUn/1997)

- **Avaliações**

Primeira avaliação: peso 4,0

Segunda avaliação: peso 4,5

Seminário: peso 1,5

- \* As provas poderão conter questões objetivas, objetivas mistas e dissertativas.

### **Observações:**

#### **Avaliação de recuperação**

- Não há avaliação de recuperação nas disciplinas de **caráter prático** que envolve atividades de laboratório (Res.17/CUn/97).

#### **Nova avaliação**

- Para pedido de segunda avaliação somente em casos em que o aluno, por motivo de força maior e plenamente justificado, deixar de realizar avaliações previstas no plano de ensino, deverá formalizar pedido de avaliação à Direção do Campus Araranguá dentro do prazo de 3 dias úteis apresentando comprovação. ([Ver formulário](#))

XI. CRONOGRAMA TEÓRICO/PRÁTICO		
AULA (semana)	DATA	ASSUNTO
1 <sup>a</sup>	05/03/12 a 10/03/12	Unidade I: Plano de Ensino e Introdução à Engenharia de Software: Visão Geral
2 <sup>a</sup>	12/03/12 a 17/03/12	Unidade I: Processos de software clássicos
3 <sup>a</sup>	19/03/12 a 24/03/12	Unidade I: Processo Unificado
4 <sup>a</sup>	26/03/12 a 31/03/12	Unidade I: <i>Extreme programming</i>
5 <sup>a</sup>	02/04/12 a 07/04/12	Unidade I: Desenvolvimento de Software
6 <sup>a</sup>	09/04/12 a 14/04/12	Unidade I: Verificação e Validação: testes
7 <sup>a</sup>	16/04/12 a 21/04/12	Unidade I: Abordagens voltada ao reuso de software
8 <sup>a</sup>	23/04/12 a 28/04/12	Unidade I: Reengenharia, evolução de sistemas legados, manutenção de software
9 <sup>a</sup>	30/04/12 a 05/05/12	Unidade II: Qualidade
10 <sup>a</sup>	07/05/12 a 12/05/12	<b>Prova Teórica Unidade I</b>
11 <sup>a</sup>	14/05/12 a 19/05/12	Unidade II: Gerenciamento de Software: projetos e requisitos
12 <sup>a</sup>	21/05/12 a 26/05/12	Unidade II: Modelos de maturidade: CMMI
13 <sup>a</sup>	28/05/12 a 02/06/12	Unidade II: Modelos de maturidade: CMMI
14 <sup>a</sup>	04/06/12 a 09/06/12	Unidade II: Modelos de maturidade: MPS.BR
5 <sup>a</sup>	11/06/12 a 16/06/12	Unidade II: Modelos de maturidade: MPS.BR
16 <sup>a</sup>	18/06/12 a 23/06/12	Unidade II: Apoio automatizado ao gerenciamento de <i>software</i>
17 <sup>a</sup>	25/06/12 a 30/06/12	<b>Prova Teórica Unidade II e Seminários</b>
18 <sup>a</sup>	02/07/12 a 07/07/12	Seminários, prova de reposição e <b>nova avaliação</b>
19a	09/07/12 a 11/07/12	Divulgação das notas finais

#### XII. Feriados previstos para o semestre 2012.1:

DATA	
03/04/2012	Aniversário de Araranguá
06/04/2012	Sexta-feira santa
30/04/2012	Dia não letivo
01/05/2012	Dia do Trabalho – Feriado Nacional (Lei nº 662/49)
04/05/2012	Dia não letivo (campus de Araranguá - Dia da Padroeira da Cidade)
07/05/2012	Corpus Christi
08/05/2012	Dia não letivo

#### XIII. BIBLIOGRAFIA BÁSICA

OMMERVILLE, I. **Engenharia de software**. 8. ed. São Paulo: Pearson Addison-Wesley, 2007. xiv, 552 p.

BECK, K. **Programação extrema (xp) explicada: acolha as mudanças**. Porto Alegre: Bookman, 2004. 182p.

GAMMA, E. ET AL. **Padrões de projeto: soluções reutilizáveis de software orientado a objetos**. Porto Alegre: Bookman, 2000. 364p

#### XIV. BIBLIOGRAFIA COMPLEMENTAR

PAULA FILHO, W. **Engenharia de software: fundamentos, métodos e padrões**. 3.ed. Rio de Janeiro: LTC, c2009.

MENDES, E.; MOSLEY, N. **Web Engineering**. New York: Springer, 2007.

LARMAN, C. **Utilizando UML e padrões: uma introdução à análise e ao projetos orientados a objetos e ao desenvolvimento interativo**. 3. ed. Porto Alegre: Bookman, 2007.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML - Guia do Usuário**. 2. ed. Rio de Janeiro: Campus, 2006.  
PRESSMAN, R. **Engenharia de software**. 6. ed. São Paulo: McGraw Hill, 2006. 752p.

JACOBSON, I; BOOCH, G.; RUMBAUGH, J. **The unified software development process**. Boston: Addison-Wesley, 1999.

Os livros acima citados constam na Biblioteca Universitária e Setorial de Araranguá. Algumas bibliografias também podem ser encontradas no acervo da disciplina, via sistema Moodle.

  
Prof.ª Luciana Bolan Frigo

Aprovado na Reunião do Colegiado do Curso 03/04/12

  
Coordenador do Curso

*Anderson Luiz Fernandes Perez, Dr.*  
Prof. Adjunto/SIAPE: 1635680  
UFSC/Campus Araranguá