



UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE COMPUTAÇÃO
PLANO DE ENSINO

SEMESTRE 2017.2

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
04652 – 4-1830-2 e 6-1830-2		

II. PROFESSOR(ES) MINISTRANTE(S)

Lucas Borges Castellan
Lucas.castellan@ufsc.br

III. PRÉ-REQUISITO(S) SOMENTE PARA CURSO TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO

CÓDIGO	NOME DA DISCIPLINA
ARA 7124	Engenharia de Software I

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

Tecnologias da Informação e Comunicação

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

Evolução 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
- Gerenciamento de projeto de *software*
- 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**
Média das avaliações: peso 7,0
Média das atividades: peso 3,0

* 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

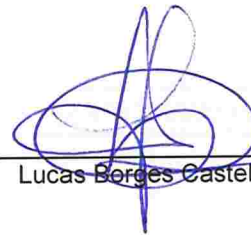
- Para pedido de segunda avaliação somente em casos em que o aluno, por motivo de força maior e plenamente justificado, deixar de realizar avaliações previstas no plano de ensino, deverá formalizar pedido de avaliação à Direção do Campus Araranguá dentro do prazo de 3 dias úteis apresentando comprovação.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

XI. CRONOGRAMA TEÓRICO/PRÁTICO		
AULA (semana)	DATA	ASSUNTO
1	30/07 a 06/08	Unidade I: Plano de Ensino e Introdução à Engenharia de Software: Visão Geral
2	07/08 a 13/08	Unidade I: Processos de software clássicos
3	14/08 a 20/08	Unidade I: Processos Ágeis
4	21/08 a 27/08	Unidade I: Processos Ágeis
5	28/08 a 03/09	Unidade I: Verificação e Validação
6	04/09 a 10/09	Unidade I: Manutenção
7	11/09 a 17/09	Unidade II: Qualidade
8	18/09 a 24/09	Prova Teórica Unidade I
9	25/09 a 01/10	Unidade II: Abordagens voltada ao reuso de software
10	02/10 a 08/10	Unidade II: Gerenciamento de Software: projetos e requisitos
11	09/10 a 15/10	Unidade II: Modelos de maturidade: CMMI
12	16/10 a 22/10	Unidade II: Modelos de maturidade: CMMI
13	23/10 a 29/10	Unidade II: Modelos de maturidade: MPS.BR
14	30/10 a 05/11	Unidade II: Modelos de maturidade: MPS.BR
15	06/11 a 12/11	Unidade II: Reengenharia, evolução de sistemas legados, Apoio automatizado ao gerenciamento de <i>software</i>
16	13/11 a 19/11	Prova Teórica Unidade II e Segunda avaliação (Prova substitutiva)
17	20/11 a 26/11	Nova Avaliação (Prova de recuperação)
18	27/11 a 03/12	Divulgação de Notas

XII. Feriados previstos para o semestre 2017.1:	
DATA	
07/09/2017	Independência do Brasil
12/10/2017	Nossa Senhora Aparecida
28/10/2017	Dia do Servidor Público
02/11/2017	Finados
25/12/2017	Natal

XIV. BIBLIOGRAFIA COMPLEMENTAR
PRESSMAN, Roger S. Engenharia de software: uma abordagem profissional . 7. ed. Porto Alegre: AMGH, 2011. 780 p.
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.
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



Lucas Borges Castellan

Aprovado na Reunião do Departamento ___/___/___

Aprovado na Reunião do Colegiado do Curso ___/___/___

Chefe de Departamento



Coordenador do Curso

Profa. Dra. Patricia Jantsch Fiuza
Coordenadora do Curso de Bacharelado em
Tecnologias da Informação e Comunicação
Centro de Ciências, Tecnologias e Saúde
Portaria nº 250/2017/GR