



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
09655 – 2-1420-2 e 4-1420-2		

**II. PROFESSOR(ES) MINISTRANTE(S)**

Luciana Bolan Frigo

**III. PRÉ-REQUISITO(S) SUGERIDO**

CÓDIGO	NOME DA DISCIPLINA
ARA 7138	Análise e Projeto de Software

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

Engenharia de Computação

**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 Secretaria Integrada de Departamento dentro do prazo de **3 dias úteis** apresentando comprovação.

<b>XI. CRONOGRAMA TEÓRICO/PRÁTICO</b>		
<b>AULA (semana)</b>	<b>DATA</b>	<b>ASSUNTO</b>
1	31/07/2017	Unidade I: Plano de Ensino
2	02/08/2017	Introdução à Engenharia de Software: Visão Geral
3	07/08/2017	Unidade I: Processos de software clássicos
4	09/08/2017	Unidade I: Processos de software clássicos
5	14/08/2017	Unidade I: Processos de software clássicos
6	16/08/2017	Unidade I: Processos de software clássicos
7	21/08/2017	Unidade I: Processos Ágeis
8	23/08/2017	Unidade I: Processos Ágeis
9	28/08/2017	Unidade I: Processos Ágeis
10	30/08/2017	Unidade I: Processos Ágeis
11	04/09/2017	Unidade I: Processos Ágeis
12	06/09/2017	Unidade I: Verificação e Validação
13	11/09/2017	Unidade I: Verificação e Validação
14	13/09/2017	<b>Prova Teórica Unidade I</b>
15	18/09/2017	Unidade II: Manutenção
16	20/09/2017	Unidade II: Manutenção
17	25/09/2017	Unidade II: Manutenção
18	27/09/2017	Unidade II: Qualidade
19	02/10/2017	Unidade II: Qualidade
20	04/10/2017	Unidade II: Qualidade
21	09/10/2017	Unidade II: Abordagens voltada ao reúso de software
22	11/10/2017	Unidade II: Abordagens voltada ao reúso de software
23	16/10/2017	Unidade II: Gerenciamento de Software: projetos e requisitos
24	18/10/2017	Unidade II: Gerenciamento de Software: projetos e requisitos
25	23/10/2017	Unidade II: Gerenciamento de Software: projetos e requisitos
26	25/10/2017	Unidade II: Modelos de maturidade: CMMI
27	30/10/2017	Unidade II: Modelos de maturidade: CMMI
28	01/11/2017	Unidade II: Modelos de maturidade: MPS.BR
29	06/11/2017	Unidade II: Modelos de maturidade: MPS.BR
30	08/11/2017	Unidade II: Modelos de maturidade: MPS.BR
31	13/11/2017	Unidade II: Modelos de maturidade: MPS.BR
32	15/11/2017	<b>FERIADO</b>
33	20/11/2017	Unidade II: Reengenharia, evolução de sistemas legados, Apoio automatizado ao gerenciamento de <i>software</i>
34	22/11/2017	<b>Prova Teórica Unidade II</b>
35	27/11/2017	<b>Segunda avaliação (P1 ou P2)</b>
36	29/11/2017	<b>Prova de recuperação</b>
37	04/12/2017	<b>Divulgação de Notas</b>

<b>XII. Feriados previstos para o semestre 2017.2:</b>	
<b>DATA</b>	
07/09	Independência do Brasil
08 e 09/09	Dias não letivos
12/10	Nossa Senhora Aparecida
13 e 14/10	Dias não letivos
02/11	Finados
15/11	Proclamação da República

#### 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.

---

Professora

Aprovado na Reunião do Colegiado do Curso \_\_\_/\_\_\_/\_\_\_

---

Coordenador do Curso