



Plano de Ensino

1. Dados de Identificação

Curso: **Ciência da Computação (Noturno)**
Componente Curricular: **Engenharia de Software I**
Fase: **Quinta**
Ano/Semestre: **2013/2**
Numero de Créditos: **4**
Carga horária - Hora Aula: **72**
Carga horária - Hora Relógio: **60**
Professora: **Raquel Aparecida Pegoraro**

2. Objetivo Geral do Curso

O curso tem por objetivo a formação integral de novos cientistas e profissionais da computação, os quais deverão possuir conhecimentos técnicos e científicos e serem capazes de aplicar estes conhecimentos, de forma inovadora e transformadora, nas diferentes áreas de conhecimento da Computação. Adicionalmente, os egressos do curso deverão ser capazes de adaptar-se às constantes mudanças tecnológicas e sociais, e ter uma formação ao mesmo tempo cidadã, interdisciplinar e profissional.

3. Ementa

Processo de desenvolvimento de software. Ciclo de vida de desenvolvimento de software. Qualidade de software. Técnicas de planejamento e gerenciamento de software. Gerenciamento de configuração de software. Engenharia de requisitos. Métodos de análise e de projeto de software. Garantia de qualidade de software. Verificação, validação e teste. Manutenção. Documentação. Padrões de desenvolvimento. Reuso. Reengenharia. Ambientes de desenvolvimento de software.

4. Justificativa

A Engenharia de Software tem um papel essencial para o desenvolvimento de produtos de software com qualidade que atendam às necessidades dos usuários. Assim, os conceitos apresentados neste componente curricular capacitam o futuro profissional da computação aplicar técnicas de engenharia no desenvolvimento de software.

5. Objetivo

5.1 Geral

- Compreender o processo de concepção, desenvolvimento, implementação e manutenção de produtos de software e os procedimentos, e as ferramentas e documentos relacionados a este processo.

5.2 Específicos

- Fornecer uma visão geral do ciclo de desenvolvimento de software;
- Compreender os processos envolvidos no desenvolvimento de sistemas;
- Entender os modelos de maturidade para desenvolvimento de sistemas no controle de qualidade;
- Conhecer os processos de engenharia de requisitos, gerência de configuração e medição de software;



Universidade Federal da Fronteira Sul

- Entender o papel da qualidade de software no ciclo de desenvolvimento.

6. Cronograma e Conteúdo Programático

Semana	Assunto
1	Apresentação do plano ensino. Introdução à engenharia de software. Crise do software Papéis no desenvolvimento de software
2	Sistemas sócio-técnicos. Ética na Engenharia de software. Ciclo de vida de software.
3	Modelos de processo de software: cascata, iterativo/incremental, espiral e RUP.
4	Requisitos de software. Requisitos de usuário e de sistema. Requisitos funcionais e não funcionais.
5	Técnicas de obtenção de requisitos. Processo de engenharia de requisitos: desenvolvimento de requisitos.
6	Processo de engenharia de requisitos: gerência de requisitos Métodos de análise e de projeto de software, e padrões.
7	Técnicas de planejamento e gerenciamento de software: Métodos tradicionais X Métodos ágeis.
8	Métodos ágeis Avaliação NP1.
9	Métricas de software. Recuperação avaliação NP1
10	Métricas de software (continuação).
11	Gerenciamento de configuração de software e ferramentas para controle de versão.
12	Arquitetura de software. Ambientes de desenvolvimento.
13	Verificação, validação e teste: modelo V e tipos de testes.
14	Verificação, validação e teste: planejamento de testes.
15	Gerenciamento da qualidade de software: qualidade do produto e do processo, modelos de maturidade CMMI e MPS.BR
16	Gerenciamento da qualidade de software: normas ISO/IEC 12207 e ISO/IEC 15504.
17	Entrega e manutenção de software. Tipos de manutenção (corretiva, adaptativa e evolutiva). Reengenharia. Processo de evolução do software.
18	Avaliação NP2. Prova de recuperação NP2.

Total:
72 h/aula

7. Procedimentos Metodológicos (estratégias de ensino, equipamentos, entre outros)

Conduzir a disciplina com aulas expositivas/dialogadas enquanto discutidos os itens de cunho



Universidade Federal da Fronteira Sul

teórico, evoluindo em tópicos específicos para exercícios práticos, demonstrações, contextualização baseada em publicações atualizadas. Uso de atividades em laboratórios com o objetivo de apresentar/exercitar os conceitos estudados.

O horário de atendimento dos estudantes será nas sextas-feiras no período vespertino.

O plágio e a cola serão tratados de forma rígida (nota 0 para os envolvidos na atividade em questão).

8. Avaliação do Processo Ensino-Aprendizagem

Uso de abordagens tais como: provas teóricas, avaliação escrita em aula, exercícios extra-classe, entre outros.

As avaliações serão agrupadas em dois momentos (conforme instrução normativa No. 001/Prograd/2010) Notas Parciais 1 e 2 (NP1 e NP2, respectivamente). A NP1 será composta por uma avaliação escrita (A1) e trabalhos (T1) pedidos durante o período da NP1 com o seguinte cálculo:

$$NP1 = (A1*0,8) + (T1*0,2)$$

A NP2 será composta por uma avaliação escrita (A2) e trabalhos (T2), seguindo o seguinte cálculo:

$$NP2 = (A2*0,8) + (T2*0,2)$$

Sendo que a média final (MF) será calculada com a fórmula:

$$MF=(NP1+NP2)/2$$

Será ofertada reposição de conteúdo e prova aos estudantes que não obtiveram média maior ou igual a 6,0 em uma das NPs. As reposições estão prevista no conteúdo programático acima.

A reposição será das avaliações e substitutiva. Os trabalhos não poderão ser recuperados.

Recuperação da NP1 será na semana subsequente a avaliação A1, e a recuperação da NP2 será na semana subsequente a avaliação A2, conforme previsto no calendário das aulas.

O cálculo da média após a prova de recuperação (PR) será feito da seguinte forma:

$$NP1 = ((A1*0,5+PR1*0,5)* 0,8) + (T1 * 0,2)$$

ou

$$NP2 = ((A2*0,5+PR2*0,5)* 0,8) + (T2 * 0,2)$$

9. Referências

9.1 Básicas

- SOMMERVILLE, I. Engenharia de Software. 8. ed. São Paulo: Addison - Wesley, 2008.
PRESSMAN, R. S. Engenharia de Software. 6. ed. São Paulo: Bookman Companhia Ed., 2006.
PFLEEGER, S. L. Engenharia de Software. 2. ed. São Paulo: Prentice-Hall, 2004.
LARMAN, C. Agile And Iterative Development. São Paulo: Prentice-Hall, 2003.

9.2 Complementares

- GARCIA-MOLINA, Hector; ULLMAN, Jeffrey D; WIDOM, Jennifer. Implementação de sistemas de banco de dados. Rio de Janeiro: Campus, 2001.
TONSIG, S. L. Engenharia de Software: Análise e Projeto de Sistemas. São Paulo: Futura, 2003.



Universidade Federal da Fronteira Sul

CARVALHO, A. M. B. R.; CHIOSSI, T. C. dos S. Introdução à Engenharia de Software. São Paulo: UNICAMP, 2001.

RIOS, E. Análise de Riscos em Projetos de Teste de Software. Rio de Janeiro: Alta Books, 2005.

BASTOS, A. Base de Conhecimento em Teste de Software. São Paulo: Martins, 2007