



Plano de Ensino

1. Dados de Identificação

Curso: Ciência da Computação (Matutino)
Componente Curricular: Engenharia de Software I
Fase: Quinta
Ano/Semestre: 2012/1
Numero de Créditos: 4
Carga horária - Hora Aula: 72
Carga horária - Hora Relógio: 60
Professor: 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, 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
- Conhecer os processos de engenharia de requisitos, gerência de configuração, medição de software
- Entender o papel da qualidade de software no ciclo de desenvolvimento



6. Cronograma e Conteúdo Programático

Semana	Aulas	Total Parc.	Assunto
1	3	3	Apresentação plano ensino Introdução à engenharia de software
2	5	8	Crise do software Metodologia e processo de software Papéis no desenvolvimento de software
3	5	13	Engenharia de Requisitos
4	5	18	Engenharia de Requisitos (requisitos funcionais e não-funcionais) (requisitos de usuário e de sistema)
5	5	23	Tipos de decomposição (objetos, eventos, processos, dados) Arquitetura de software Primeira avaliação A1 (quinta-feira)
6	5	28	Testes de software Teste caixa branca e preta
7	5	33	Tipos de testes (componente, sistema e aceitação)
8	5	38	Planejamento de testes Segunda avaliação A2 (quinta-feira)
9	5	43	Manutenção de software Tipos de manutenção (corretiva, adaptativa e evolutiva) Processo de evolução do software
10	5	48	Gerência de configuração: item de configuração, baseline, versionamento, trunk e branch
11	5	53	Ferramentas (Git, SVN) Estimativa de software
12	5	58	Estimativa de software: conceitos de tamanho de software, pontos por função, use case points, tamanho nas metodologias ágeis
13	5	63	Técnicas de planejamento e gerenciamento de software
14	5	68	Qualidade de software Modelos de maturidade
15	4	72	Terceira avaliação A3
		72	

* O plano e cronograma podem ser alterados pelo professor ao longo do semestre.

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

Conduzir a disciplina com aulas expositivas/dialogadas enquanto discutidos os itens de cunho 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, trabalhos de implementação, 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:



$$NP1=A1*0,7+T1*0,3$$

A NP2 será composta por duas avaliações escritas (A2 e A3) e trabalhos (TF), seguindo o seguinte cálculo:

$$NP2=(A2+A3)/2*0,7+TF*0,3$$

Sendo que a média final (MF) será calculada como $MF=(NP1+NP2)/2$

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 no período vespertino.

Recuperação da NP2 será na semana subsequente a avaliação A3 no período vespertino.

O cálculo da média após a recuperação será feito da seguinte forma:

$NPn*0,4+NR * 0,6$, onde NPn é a média da NP1 e/ou NP2 e NR é a nota da recuperação.

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.

9.1 Específicas

- 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.
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