



Plano de Ensino

1. Dados de Identificação

Curso: **Ciência da Computação (Matutino)**
Componente Curricular: **Engenharia de Software II**
Fase: **Sexta**
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

Análise de Requisitos. Técnicas existentes para a extração de requisitos. Modelos de Processos existentes para o desenvolvimento de aplicações orientadas a objetos. O paradigma orientado a objetos. Técnicas de análise orientada a objetos. Utilização de uma ferramenta de apoio para a elaboração dos modelos de análise.

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 as diversas técnicas de levantamento de requisitos e métodos de análise. Analisar e projetar sistemas informatizados utilizando uma metodologia orientada a objetos.

5.2 Específicos

- Compreender as diferentes técnicas existentes para extração de requisitos;
- Conhecer os modelos de processos existentes para o desenvolvimento de aplicações orientadas a objetos;
- Conhecer o método de análise e projeto orientado a objetivos utilizando UML;
- Utilizar uma ferramenta de apoio para a elaboração dos modelos da UML;
- Elaborar trabalho prático de desenvolvimento de projeto de software de software, onde



o aluno possa vivenciar a prática necessária para entender os conteúdos estudados.

6. Cronograma e Conteúdo Programático

Semana	Assunto
1	Apresentação do plano ensino. Visão geral da UML. Processo de engenharia de requisitos.
2	Técnicas de elicitação de requisitos: diagrama de casos de uso. Ferramenta de modelagem UML Astah Community.
3	Especificação/documento de requisitos através de descrição dos casos de uso. Validação de requisitos.
4	O paradigma OO e a UML. RUP - Rational Unified Process. Diagrama de classes: classes, atributos, métodos, tipos de relacionamentos e associações.
5	Diagrama de classes (continuação).
6	Diagrama de objetos. Modelo conceitual X modelo de domínio.
7	Diagrama de pacotes.
8	Trabalho prático. Avaliação NP1
9	Diagrama de sequencia. Recuperação NP1.
10	Diagrama de sequencia (continuação)
11	Diagrama de comunicação. Diagrama de máquina de estados.
12	Diagrama de atividades. Visão geral dos outros diagramas da UML.
13	Padrões de projeto.
14	Projeto navegacional. Noções de usabilidade e critérios ergonômicos. Avaliação NP2.
15	Apresentação dos trabalhos. 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 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 um trabalho (T2), seguindo o seguinte cálculo:

$$NP2 = (A2*0,7) + (T2*0,3)$$

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

$$NP2 = ((A2*0,5+PR2*0,5)* 0,7) + (T2 * 0,3)$$

9. Referências

9.1 Básicas

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML Guia do Usuário**. Rio de Janeiro: Campus, 2006.

WAZLAWICK, R. S. **Análise e Projeto de Sistemas de Informação Orientados a Objetos**. Rio de Janeiro: Campus, 2004.

GUEDES, G. T. A. **UML 2 - Uma Abordagem Prática**. São Paulo: Novatec, 2009.

RUMBAUGH, J.; BLAHA, M. **Modelagem e Projetos Baseados em Objetos**. Rio de Janeiro: Campus, 2004.

9.2 Complementares

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.



Universidade Federal da Fronteira Sul

PFLEEGER, S. L. **Engenharia de Software**. 2. ed. São Paulo: Prentice-Hall, 2004.

LARMAN, C. **Utilizando UML e Padrões**: uma Introdução à Análise e ao Projeto Orientados a Objetos. 3. ed. São Paulo: Borkman Companhia, 2007.