

UNIDADE CURRICULAR: Programação Orientada a Objetos II		
Professor Responsável: Fábio Fagundes Silveira		Contato: fsilveira@unifesp.br
Ano Letivo: 2010	Semestre: primeiro	Pré-requisitos: Programação Orientada a Objetos I
Departamentos/Disciplinas participantes:		
Carga horária total: 72 horas		
Carga Horária p/ prática (em %): 50		Carga Horária p/ teoria (em %): 50
<p>Objetivos: O objetivo dessa disciplina é apresentar tópicos avançados sobre programação orientada a objetos, visando a sedimentar os conhecimentos em programação dos estudantes, fornecendo-lhes uma visão mais geral sobre linguagens de modelagem OO, padrões de projeto, arquiteturas três camadas e métodos alternativos de desenvolvimento de software, como por exemplo, Programação Orientada a Aspectos e Métodos Ágeis. Ao final do curso, os alunos deverão ser capazes de desenvolver sistemas mais complexos em equipe, utilizando linguagens de modelagem e recursos avançados de programação, bem como fazer uso de ambientes e ferramentas de desenvolvimento.</p>		
<p>Ementa: Modelagem Orientada a Objetos Utilizando UML e SysML; Padrões de Projeto; Arquiteturas de três camadas; Métodos Ágeis; Refatoramento; Estudos de Caso.</p>		
<p>Conteúdo Programático:</p> <ol style="list-style-type: none"> 1. Linguagem de Modelagem Unificada (UML) <ol style="list-style-type: none"> a. Histórico e Introdução b. Casos de Uso c. Requisitos com casos de uso d. Diagramas da UML: casos de uso, atividades, classes, seqüência, colaboração ou comunicação, objetos, estados, componentes, implantação, tempo e. Técnicas de análise orientada a objetos: identificação de classes, relacionamentos, operações e atributos. f. Estudo de Caso e Exercícios 2. Linguagem de Modelagem de Sistemas (SysML) <ol style="list-style-type: none"> a. Histórico e Introdução b. Descrição e Arquitetura da Linguagem c. Especificação da SysML d. Diagramas da SysML e. Estudo de Caso e Exercícios 3. Arquitetura de três camadas <ol style="list-style-type: none"> a. Introdução b. Camada de interface c. Camada de negócios d. Camada de dados e. Estudo de Caso e Exercícios 		

4. Métodos Ágeis
 - a. Introdução
 - b. Principais Métodos Ágeis: XP, SCRUM, outros.
 - c. Elaboração progressiva de requisitos
 - d. Planejamento Iterativo e adaptação.
5. Refatoramento
 - a. Introdução
 - b. Princípios
 - c. “*Bad Smells*”
 - d. Ferramentas de refatoração
 - e. Estudos de Caso e Exercícios
6. Programação Orientada a Aspectos
 - a. Introdução à Separação de Interesses
 - b. Aspectos
 - c. JoinPoints, Pointcuts e Advices

Metodologia de Ensino Utilizada:

A disciplina será intercalada por aulas teóricas e aulas práticas em laboratório. Nas aulas teóricas serão apresentados os principais conceitos e seus relacionamentos; nas aulas de laboratório, os conceitos serão implementados em ferramentas CASE e em linguagem Java. Ademais, desenvolver-se-á atividades à distância, com o apoio da ferramenta *Moodle*. Um monitor será contratado para dar apoio pedagógico aos alunos.

A metodologia de ensino baseada na resolução de problemas (*Problem Based Learning*) será amplamente utilizada. O professor, após apresentar a teoria necessária, irá propor problemas e atuará apenas como facilitador junto aos alunos na resolução do problema.

Recursos Instrucionais Necessários:

- Projetor multimídia;
- Laboratório com 50 computadores equipados com o sistema operacional Linux, linguagem Java e ferramenta CASE;
- Sala de aula com quadro branco;
- Ambiente de apoio à aprendizagem colaborativa à distância (*Moodle*);
- Monitor da disciplina com carga horária de cinco horas por semana.

Avaliação:

A avaliação do aprendizado dar-se-á por meio de uma ou duas provas semestrais (i), trabalhos individuais (ii), trabalhos em grupo (iii) e lista de exercícios (iv).

A nota final será uma média ponderada entre as notas dos itens i (peso 3), ii (peso 1), iii (peso 1) e iv (peso 1).

Bibliografia básica:

- VLISSIDES, J., Helm, H., GAMMA, E, JOHNSON, R., *Padrões de Projeto*, Editora Bookman, 2005.
- WAZLAWICK, R. S. *Análise e Projeto de Sistemas de Informação Orientados a Objetos*, Editora Campus, 2004.

- BEZERRA, E. *Princípios de Análise e Projeto de Sistemas com UML*. 2ª ed. Rio de Janeiro: Elsevier, 2007
- BUSHMANN, F., STAL, M., Meunier, R., SOMMERLAD, P., *Pattern- Oriented Software Architecture: a System of Patterns*. Editora Wiley, 1996.
- LARMAN, C.; Salvador, L.M.A, *Utilizando UML e Padrões: Uma Introdução à Análise e ao Projeto Orientado a Objetos*. Porto Alegre: Bookman, 2000.
- LADDAD, R. AspectJ in action: practical aspect-oriented programming. Greenwich: Manning Publications Co., 2003. ISBN 1930110936.
- GRADY BOOCH; JAMES RUMBAUGH; IVAR JACOBSON. *UML: Guia do Usuario*. 2ª ed. Rio de Janeiro: Elsevier, 2005.
- GAMMA, E.; HELM, R. *Design Patterns: Elements Of Reusable Object-Oriented Software*. Boston: Addison-Wesley, 2002.

Docentes Participantes

Nome	Origem (Departamento)	Titulação (graduado, mestre, doutor, livre docente)	Regime de Trabalho	Carga horária (na unidade)
Fábio Fagundes Silveira	DCT	Doutor	DE	40 horas