

PLANO DE ENSINO

1. IDENTIFICAÇÃO:

CURSO: Pós Graduação Lato Sensu em Engenharia de Software	SEMESTRE/ANO: 2º / 2011
DISCIPLINA: Análise e Projeto Orientado a Objetos - UML.	PERÍODO MINISTRADO:
PROFESSOR:	CARGA HORÁRIA: 40 horas aula

TURMA(S)/HORÁRIO:

2. EMENTA:

Conceito sobre orientação a objetos: classe e objeto, herança, polimorfismo, troca de mensagens, associação entre objetos. Projeto de software orientado a objetos. Padrões de projeto. Arquitetura de software orientada a objetos. Componentes de software. Programação Orientada a Objetos: implementação de classes, modos de encapsulamento, herança, comunicação entre objetos, interfaces e *threads*.

3. CONTRIBUIÇÃO DA DISCIPLINA PARA OS OBJETIVOS DO CURSO:

Disseminar o conhecimento sobre a cultura de análise e projeto orientados a objetos (APOO) com UML em seus diversos níveis e mostrar os benefícios da APOO como prática efetiva da Engenharia de Software. Identificar os desafios enfrentados na institucionalização de um processo de APOO e os ganhos obtidos para a melhoria na qualidade do processo de desenvolvimento do produto de software.

4. OBJETIVO GERAL:

A disciplina pretende capacitar o aluno a utilizar o paradigma da análise e projeto orientados a objetos, a linguagem UML e o processo unificado para conduzir as fases de análise e projeto de sistemas de informações.

5. OBJETIVOS ESPECÍFICOS:

- ✓ Apresentar uma visão sobre principais aspectos da orientação a objetos.
- ✓ Capacitar o aluno no desenvolvimento de software orientado a objetos, envolvendo análise, projeto e implementação de software.
- ✓ Capacitar o aluno no uso de métodos e técnicas de modelagem e implementação de software orientado a objetos.
- ✓ Discutir os principais benefícios e restrições do paradigma da programação orientada a objetos.

6. CONTEÚDO PROGRAMÁTICO:

Unidade 1. Introdução a Orientação a Objetos.

- ✓ 1. História da Análise Orientada a Objetos (evolução) 2. Conceitos Básicos de Orientação a Objetos (classes, objetos, atributos, operações, herança, abstração, representação, encapsulamento, reuso e comunicação entre objetos).

Unidade 2. O processo de desenvolvimento de Software.

- ✓ 1. O Ciclo do Desenvolvimento de Software Orientado a Objetos. 2. O Processo Unificado. Modelagem da Arquitetura de um Sistema. 3. Visão Geral da UML: Vocabulário da UML, Diagramas, Regras e Mecanismos Comuns (notas, estereótipos, valores atribuídos).

Unidade 3. Artefatos usados pelo Processo Unificado.

- ✓ 1. Casos de Uso: Diagrama de Caso de Uso. 2. Modelo Conceitual: Diagramas de Seqüência, Atividades, Execução e Componentes, Contratos, Diagrama de Classes, Diagrama de Interação. Desenvolvimento de Estudos de Caso em uma Ferramenta de Modelagem.

Unidade 4. Programação Orientada a Objetos.

- 1. implementação de classes 2. modos de encapsulamento 3. herança 4. comunicação entre objetos. 5. interfaces e *threads*.

6 .BIBLIOGRAFIA (básica e complementar):

Bibliografia básica

1. BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro: Elsevier, 2002. 286 p.
2. LARMAN, Craig. **Utilizando UML e padrões: uma introdução à análise e ao projetos orientados a objetos e ao processo unificado**. 3ª ed. Porto Alegre: Bookman, 2007. 607 p.
3. PENDER, Tom; VIEIRA, Daniel. **UML a bíblia**. Rio de Janeiro: Elsevier, 2004. 711 p.
4. QUATRANI, Terry. **Modelagem visual com rational rose 2000 e UML**. Rio de Janeiro: Ciência Moderna, 2001. 206 p.
5. FOWLER, Martin. **UML Essencial**. 3ª ed. Porto Alegre: Bookman, 2004. 160p

Bibliografia complementar

1. DEITEL, H. M. **Java, como Programar**. Bookman, 2003.
2. LARMAN, C.; **Utilizando UML e Padrões**, Bookman Editora, 2ª Edição, 2002.
3. JACOBSON, I. BOOCH, G., RUMBAUGH, J.. **The Unified Software Development Process**, Addison-Wesley Object Technology Series, 1999.
4. LADD, S. R. **Java Algorithms**. McGraw-Hill, 1998.
5. MARTIN, F. **UML Distilled - Applying the Standard Object Modeling Language**. Addison-Wesley, 1999.
6. O'DOCHERTY, M. **Object-Oriented Analysis and Design**. Wiley, 2005.
7. SAVITCH, W. J. **C++ Absoluto**, Addison-Wesley, 2004.
8. SHILDT, H. Borland **C++: Completo e Total**, Makron Books, 1998.
9. SIKORA, Z. M. **Java: Guia Prático para Programadores**. Ed. Campus, 2003.
10. WEISFELD, M. **The Object-Oriented Thought Process**, 2nd edition, 2003.