



## Ficha 2 (variável)

<b>Disciplina:</b> Design de Software						<b>Código:</b> CI1163	
<b>Natureza:</b> ( X ) Obrigatória ( ) Optativa			( X ) Semestral ( ) Anual ( ) Modular				
<b>Pré-requisito:</b> CI1162,CI1062		<b>Co-requisito:</b>		<b>Modalidade:</b> ( X ) Presencial ( ) Totalmente EAD ( ) ..... % EAD <sup>1</sup>			
<b>CH Total:</b> 60 <b>CH semanal:</b> 4	<b>Padrão(PD):</b> PD=30	<b>Laboratório(LB):</b> LB=30	<b>Campo(CP)</b> CP=0	<b>Estágio(ES)</b> ES=0	<b>Orientada(OR):</b> OR=0	<b>Prática Específica(PE):</b> PE=0	
<b>EMENTA (Unidade Didática)</b>							
Design de software. Interação entre análise de requisitos e design. Modelos de design de software. Linguagem de modelagem. Design da arquitetura de software. Design detalhado. Construção de modelos de design de software. Padrões de design. Revisões e inspeções. Correspondência entre design e codificação.							
<b>PROGRAMA (itens de cada unidade didática)</b>							
1. Fundamentos de Design de Software 2. Arquitetura de Software (a) Estilos Arquiteturais; (b) Padrões Arquiteturais; 3. Modelos de Design de Software Geral e Detalhado (Modelos UML) (a) Modelos Comportamentais; (b) Modelos Estruturais; 4. Do Design para a codificação 5. Padrões de Design de Software							
<b>OBJETIVO GERAL</b>							
Ser capaz de projetar e modelar (desenhar) soluções de software passíveis de codificação, a partir de problemas modelados em etapa anterior.							
<b>OBJETIVO ESPECÍFICO</b>							
1. O aluno deverá conhecer as principais técnicas de Design de Software; 2. O aluno deverá ser capaz de construir os modelos de Design tanto geral quanto detalhado; 3. O aluno deverá ser capaz de aplicar padrões de Design de software.							
<b>PROCEDIMENTOS DIDÁTICOS</b>							
Aulas teóricas expositivas dialogadas ministradas por professores do Departamento de Informática em anfiteatros/salas de aula, utilizando-se os recursos de quadro de giz e projetores. Aulas práticas em laboratório de Informática.							
<b>FORMAS DE AVALIAÇÃO</b>							



Ministério da Educação  
Universidade Federal do Paraná  
Setor de Ciências Exatas  
Departamento de Informática

Trabalhos práticos, provas e seminários.

#### **BIBLIOGRAFIA BÁSICA**

1. Craig Larman. Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo. Porto Alegre: Bookman, 2007.
2. Oliver Vogel. Software architecture: a comprehensive framework and guide for practitioner. Berlin: Springer, 2011.
3. Ralph Johnson Erich Gamma Richard Helm e John Vlissides. Design patterns : elements of reusable object-oriented software. Reading, Mass. : Addison-Wesley, 1995.

#### **BIBLIOGRAFIA COMPLEMENTAR**

1. Eduardo Bezerra. Princípios de análise e projeto de sistemas com UML. Rio de Janeiro : Elsevier, 2015.
2. Eric J. Braude. Projeto de software: da programação à arquitetura : uma abordagem baseada em Java. Porto Alegre: Bookman, 2005.
3. Grady Booch, James Rumbaugh e Ivar Jacobson. UML : guia do usuário : o mais avançado tutorial sobre Unified Modeling Language (UML), elaborado pelos próprios criadores da language. Rio de Janeiro: Campus: Elsevier, 2012.
4. Ian Sommerville. Engenharia de software. São Paulo : Pearson Prentice Hall, 2011.
5. Roger S. Pressman. Engenharia de software: uma abordagem profissional. Porto Alegre (RS): AMGH, 2011.

**Professor da Disciplina:** Marcos Didonet Del Fabro

**Assinatura:** \_\_\_\_\_

**Chefe de Departamento:** Prof. Fabiano Silva

**Assinatura:** \_\_\_\_\_

*OBS (1): ao assinalar a opção % EAD, indicar a carga horária que será à distância.*