

Disciplina: **P03654/TIDD 5118 - Conceitos Fundamentais de Modelagem de Software (CFMS)**

Módulo: 1

Área de Concentração: Processos Cognitivos e Ambientes Digitais

Linha de Pesquisa: Modelagem de Sistemas de Software

Professor: Dr. Ítalo Santiago Vega (cód. 5078)

Semestre: 1º semestre de 2014

Horário: 3ª feira, 15h00 às 18h00

Créditos: 3

Carga Horária: 255 horas

Nível: Mestrado

Tipo: eletiva

1. Descrição e ementa da disciplina

A disciplina explora e discute campos disciplinares específicos de modo a desenvolver as seguintes competências do aluno: uso e aplicação de ferramentas, técnicas e princípios de refinamento de arquiteturas de software em modelos de implementação. Para atingir tais objetivos serão utilizadas aulas expositivas, debates sobre temas específicos, estudo de casos e exercícios, contemplando a seguinte ementa:

Complexidade, abstração e modelagem. Paradigma de objetos. Notações. Princípios e técnicas de modelagem. Padrões de desenho. Desenho de componentes. Processos. Estudos de caso.

2. Detalhamento da ementa da disciplina em unidades de conteúdo para 17 semanas de aula.

Aula 01:

Modelos estáticos e dinâmicos: Martin (2002).

Aula 02:

Elementos de modelagem em UML: Booch (2007).

Aula 03:

Desenho de modelos estáticos (i): Martin (2002).

Aula 04:

Desenho de modelos estáticos (ii): Martin (2002).

Aula 05:

Desenho de modelos dinâmicos (i): Martin (2002).

Aula 06:

Desenho de modelos dinâmicos (ii): Martin (2002).

Aula 07:

Estudo de Caso: Parte I.

Aula 08:

Classificação de objetos de Jacobson: Jacobson (1992).

Aula 09:

Estilos arquiteturais e direcionadores de desenho: Booch (2007).

Aula 10:

Gerência da complexidade de software: Page-Jones (2000).

Aula 11:

Estudo de Caso: Parte II.

Aula 12:

Descoberta e invenção de abstrações: Martin (2002), Gamma et al. (1995).

Aula 13:

Gerência de dependências: Martin (2002), Page-Jones (2000).

Aula 14:

Estudo de Caso: Parte III.

Aula 15:

Sintomas de degradação de desenho (i): Martin (2002).

Aula 16:

Sintomas de degradação de desenho (ii): Martin (2002).

Aula 17:

Estudo de caso: Parte IV.

3. Bibliografia

3.1. Básica

Booch, G. (2007). *Object-Oriented Analysis and Design with Applications*. Addison-Wesley Professional, 3rd. edition. ISBN 978-0201895513.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns Elements of Reusable Object-Oriented Software*. Addison-Wesley. ISBN 0201633612.

Jacobson, I. (1992). *Object Oriented Software Engineering: A Use Case Driven Approach*. Addison-Wesley Professional. ISBN-13: 978-0201544350.

Martin, R. C. (2002). *Agile Software Development: Principles, Patterns, and Practices*. Prentice Hall. ISBN 978-0135974445.

Page-Jones, M. (2000). *Fundamentals of Object-Oriented Design in UML. Object Technology*. Addison-Wesley. ISBN: 0-201-69946-X.

3.2. Adicional

Blaha, M. R. and Rumbaugh, J. R. (2004). *Object-Oriented Modeling and Design with UML*. Prentice Hall, 2nd edition. ISBN 978-0130159205.

Horstmann, C. (2006). *Object-oriented design and patterns*. John Wiley & Sons, 2 edition. ISBN 0-471-74487-5.

Larman, C. (2004). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Prentice-Hall, 3rd edition. ISBN 0131489062.

Martin, R. C. (2003). *UML for Java Programmers*. Prentice Hall. ISBN 978-0131428485.

McConnell, S. (2004). *Code Complete*. Microsoft Press, 2nd edition. ISBN 978-1556154843.

OBS.: Em face da natureza dinâmica do conteúdo da disciplina, outras fontes bibliográficas poderão ser sugeridas pelo professor no decorrer do semestre.

4. Avaliação

Ao longo do curso o aluno será avaliado por meio de atividades individuais ou em grupo realizadas em classe ou extra-classe.

