

Programa de Ensino

1) Identificação

Disciplina: INE5320 - Análise e Projetos de Sistemas Computadorizados II
Carga horária: 72 horas-aula Teóricas: 0 Práticas: 72
Período: início da oferta da disciplina até a presente data

2) Cursos

- Ciências da Computação (208)

3) Requisitos

- Ciências da Computação (208)
 - INE5319 - Análise e Projetos de Sistemas Computadorizados I

4) Ementa

Prática em análise e projeto de sistemas computadorizados. Análise e crítica de sistemas já desenvolvidos.

5) Objetivos

Geral: Exercitar e aperfeiçoar a prática de análise e projeto de sistemas.

Específicos:

- Exercitar análise de requisitos.
- Exercitar modelagem conceitual.
- Exercitar prototipação.
- Exercitar projeto de interfaces.
- Exercitar o uso de ferramentas CASE.
- Exercitar análise de riscos.

6) Conteúdo Programático

- 6.1) Visão Geral de Sistema. [8 horas-aula]
 - Identificação de Objetivos.
 - Modelagem de negócio.
 - Identificação de opções tecnológicas e de projeto.
- 6.2) Análise de Riscos. [12 horas-aula]
 - Identificação de riscos.
 - Classificação de riscos.
 - Elaboração de planos de redução de riscos.
- 6.3) Prototipação. [12 horas-aula]
 - Planejamento de protótipos.
 - Construção de protótipos.
 - Avaliação de protótipos.
- 6.4) Análise de Requisitos. [16 horas-aula]
 - Identificação de requisitos funcionais e não funcionais.
 - Identificação de requisitos suplementares.
 - Classificação e organização de requisitos.

- 6.5) Modelagem Conceitual. [12 horas-aula]
 - Identificação de classes e atributos candidatos.
 - Organização das informações.
 - Validação do modelo conceitual.
- 6.6) Projeto. [12 horas-aula]
 - Projeto de interface.
 - Projeto lógico.

7) Bibliografia Básica

- Wazlawick, R. S. (2010). Análise e Projeto de Sistemas de Informação Orientados a Objetos. 2a. Edição. Elsevier, Rio de Janeiro.

8) Bibliografia Complementar

- Alford, M. Requirements-driven software design. McGraw Hill, 1991.
- Ambler, S. Process patterns. Cambridge University Press, 1998.
- Ambler, S., Constantine, L., Smith, R. The Unified Process elaboration phase: best practices in implementing the UP. CMP Books, 2000.
- Arlow, J., Neustadt, I. UML and the Unified Process: practical object-oriented analysis and design. Pearson Education, 2001.
- Beck, K. Programação extrema  XP explicada: acolha as mudanças. Porto Alegre: Bookman, 2004. (Tradução de Extreme programming explained: embrace change.)
- Bezerra, E. Princípios de análise e projeto de sistemas com UML. Campus-Elsevier, 2003.
- Boehm, B.W. Software engineering. IEEE. Transactions on Computers, vol. 25, n.º 12, 1976.
- Booch, G. Object-oriented analysis and design with applications. Addison-Wesley, 1993.
- Booch, G. Object solutions  managing the object-oriented project. Addison-Wesley, 1996.
- Brown, A.W. Large-scale component-based development. Prentice-Hall, 2000.
- Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., Matera, M. Designing data-intensive Web applications. Morgan Kaufmann Publishers, 2003.
- Cox, B. Object-oriented programming: an evolutionary approach. Addison-Wesley, 1986.
- D'Souza, D.F., Wils A.C. Objects, components, and frameworks with UML. Addison-Wesley, 1999.
- Embley, D.W., Kurtz, B.D., Woodfield, S.N. Object-oriented systems analysis: a model-driven approach. Prentice-Hall, 1992.
- Erickson, H.E., Penker, M. UML toolkit. John Wiley and Sons Inc., 1998.
- Fayad, M.E., Schmidt, D.C., Johnson, R.E. Implementing application frameworks. John Wiley & Sons, Inc., 1999.
- Fowler, M., Scott, K. UML distilled. Addison-Wesley, 1997.
- Fowler, M. Patterns of enterprise application architecture. Addison-Wesley, 2003.
- Gamma, E., Helm, R., Johnson, R., Vlissides, J. Design patterns. Elements of reusable object-oriented software. Addison-Wesley, 1995.
- Jacobson, I., Christerson, M., Jonsson, P., Övergaard, G. Object-oriented software engineering- a use CASE driven approach. Addison-Wesley, 1992.
- Jacobson, I. The object advantage- business process reengineering with object technology. Addison-Wesley, 1994.
- Jacobson, I., Booch, G., Rumbaugh, J. The unified software development process. Addison-Wesley, 1999.
- Kehoe, R., Jarvis, A., Shah-Jarvis, A. Iso 9000-3: a tool for software product and process improvement. Springer Verlag, 1996.
- Larman, C. Applying UML and patterns: an introduction to object-oriented analysis and design and the unified process. Prentice Hall, 2001.
- Karner, G. Use CASE points  resource estimation for objectory projects. Objective Systems, 1993.

- Kruchten, P. The rational unified process: an introduction. Addison-Wesley, 2000.
- Kruchten, P. The rational unified process made easy: a practitioner's guide to rational unified process. Addison-Wesley, 2003.
- Maldonado, J.C., Delamaro, M.E., Jino, M. Introdução ao teste de software. Campus-Elsevier, 2007.
- Martin, R.C. Agile software development, principles, patterns, and practices. Prentice-Hall, 2002.
- Meyer, B. Object-oriented software construction. Prentice Hall, 1988.
- Meyer, B. Eiffel: the language. Prentice-Hall, 1992.
- Mitchel, R., McKim, J. Design by contract by example. Addison-Wesley, 2001.
- Object Management Group (OMG) Object Constraint Language OMG available specification version 2.0. Disponível em <http://www.omg.org/technology/documents/formal/ocl.htm>. Consultado em 26 de agosto de 2009.
- Object Management Group, OMG Unified Modeling Language  UML. Disponível em http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML. Consultado em 23 de setembro de 2009.
- Page-Jones, M. Fundamentos do desenho orientado a objeto com UML. Makron Books, 2001. (Tradução de Fundamentals of object-oriented design in UML.)
- Paula Filho, W.P. Engenharia de software: fundamentos, métodos e padrões. LTC, 2003).
- Pereira e Silva, R. UML 2  Modelagem orientada a objetos. Visual Books, 2007.
- Pereira e Silva, R. Como modelar com UML 2. Visual Books, 2009.
- Pressman, R.S. Software engineering: a practitioner's approach. McGraw Hill, 2010.
- Riel, A. J. Object-oriented design heuristics. Addison-Wesley, 1996.
- Rocha, A.R.C., Maldonado, J.C., Weber, K.C. Qualidade de software: teoria e prática. Pearson─Prentice-Hall, 2001.
- Rumbaugh, J., Blaha, M.R., Lorensen, W., Eddy, F., Premerlani, W. Object-oriented modeling and design. Prentice Hall, 1990.
- Rumbaugh, J., Jacobson, I., Booch, G. The Unified Modeling Language reference manual. Addison-Wesley, 1999.
- Santos, C.C. Geração automática de diagramas de comunicação a partir de contratos OCL. Dissertação de Mestrado, UFSC-PPGCC, 2007.
- Scott, K. The Unified Process explained. Addison-Wesley Pub Co., 2001 (O processo unificado explicado. Bookman, 2003.)
- Shalloway, A., Trott, J.R. Explicando padrões de projeto: uma nova perspectiva em projeto orientado a objeto. Bookman. (Tradução de Design patterns explained: a new perspective on object-oriented design.)
- Silva, A.A., Gomide, C.F., Petrillo, F. Metodologia e projeto de software orientados a objetos: modelando, projetando e desenvolvendo sistemas com UML e componentes distribuídos. Érica, 2003.
- Warmer, J., Keppe, A. The Object Constraint Language: precise modeling with UML. Addison-Wesley Pub Co., 1998.
- Wirfs-Brock, R., McKean, A. Object design: roles, responsibilities, and collaborations. Addison-Wesley, 2002.