

DADOS DO COMPONENTE CURRICULAR
Componente Curricular: Análise de Projeto e Software
Curso: Técnico em Informática
Período: Quarto
Carga Horária: 67 h.r
Docente: Rômulo Costa de Menezes Junior

EMENTA
<p>Fundamentos da Engenharia de Software. Análise de requisitos. Processos e metodologias de desenvolvimento de software. Padrões de notação e de modelagem. Projeto arquitetural. Mapeamento de Processos. Evolução e manutenção de software. Integração de sistemas.</p>

OBJETIVOS DE ENSINO
<p>Geral</p> <ul style="list-style-type: none"> • Permitir o aprendizado e importância da análise e projeto de software, assim como as práticas de desenvolvimento, manutenção e evolução. <p>Específicos</p> <ul style="list-style-type: none"> • Compreender o que é Engenharia de Software; • Reconhecer a importância da análise e especificação de requisitos; • Aplicar técnicas para modelagem estrutural e comportamental de sistemas; • Analisar e construir projetos arquiteturais. Compreender e aplicar técnicas para mapeamento de processos; • Conhecer a importância da manutenção, evolução e integração de softwares.

CONTEÚDO PROGRAMÁTICO
<p>UNIDADE I - Introdução a Engenharia de Software.</p> <p>1.1. O que são sistemas de software, sistemas sociotécnicos, sistemas críticos.</p> <p>1.2. Modelos de processos de software.</p> <p>1.3. Iteração de processo.</p> <p>1.4. Atividades de processo.</p> <p>1.5. O Rational Unified Process e o desenvolvimento assistido por computador.</p> <p>1.6. Desenvolvimento ágil de software.</p> <p>UNIDADE II - Requisitos de Software.</p> <p>2.1. Requisitos funcionais e não-funcionais.</p> <p>2.2. Processos de Engenharia de Requisitos.</p> <p>2.3. Especificação de sistemas de software.</p>

- 2.4. Modelos de Sistemas.
- 2.5. Casos de Uso e estórias do usuário.

UNIDADE III - Projeto de Arquitetura de Software.

- 3.1. Aspectos Gerais da Arquitetura de Software, Decisões de projeto, organização de sistema, estilos arquiteturais, visões de arquitetura.
- 3.2. Camadas de software: Camada de Interface com o Usuário, Camada de Negócio, Mapeamento Objeto-Relacional, Camada de Persistência, etc.
- 3.3. Padrões arquiteturais.

UNIDADE IV - Desenvolvimento de Software.

- 4.1. Desenvolvimento Rápido de software.
- 4.2. Reuso de Software: Padrões de Projeto, Frameworks, Reuso de sistemas e aplicações, etc.
- 4.3. Engenharia de Software Orientada a Serviços: SOAP, REST, etc.
- 4.4. Evolução de software: Dinâmica da evolução de software, Manutenção de software, Refatoração, etc.
- 4.5. Sistemas de controle de versão
- 4.6. Automação do processo de desenvolvimento

METODOLOGIA DE ENSINO

Aulas teórico-expositivas utilizando recursos audiovisuais e complementados por estudo individual e em grupo a partir do material disponibilizado; Estudos dirigidos e projetos orientados pelo professor. Recomenda-se o uso de projetos de software de código aberto (Open-Source) reais para a realização da prática dos conceitos estudados.

RECURSOS DIDÁTICOS

Como recursos didáticos serão utilizados quadro branco, pincel atômico, computador e projetor multimídia;
Laboratório com computadores individuais com desempenho adequado para uso dos softwares necessários para desenvolvimento.

PROCEDIMENTOS AVALIATIVOS

Uso de avaliação contínua e por competências ao final da disciplina. Sugestão de listas de exercício, avaliações teóricas e práticas, preferencialmente práticas, com projeto envolvendo técnicas e ferramentas apresentadas.

BIBLIOGRAFIA

Básica

- SOMMERVILLE, I. Engenharia de Software. 9. ed. São Paulo: Pearson / Prentice Hall, 2011.
- PRESSMAN, R. Engenharia de Software: Uma Abordagem Profissional. 7. ed. São Paulo: Bookman, 2011.
- MCLAUGHLIN B.; POLLICE, G.; WEST, D. Use a Cabeça! Análise e Projeto Orientado a Objetos. Rio de Janeiro: Alta Books, 2007.
- Silveira P. e outros. Introdução à Arquitetura e Design de Software: Uma visão sobre a plataforma Java. São Paulos, Brasil: Editora Campus, 2011.

Complementar

- LARMAN, C. Utilizando UML e padrões: uma introdução à análise e projeto orientado a objetos e ao desenvolvimento iterativo. São Paulo: Bookman, 2007.
- TELES, V.; Extreme Programming: Aprenda como Encantar seus Usuários Desenvolvendo software com Agilidade e Qualidade. São Paulo, Brasil: Novatec, 2004.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: Guia do usuário. 2. ed. Rio de Janeiro: Elsevier Campus, 2006.
- Fowler, M. Refatoração: Aperfeiçoando o Projeto de Código Existente. São Paulo, Brasil, Editora Bookman. 2000.
- Rafael Prikladnicki e outros. Métodos Ágeis para Desenvolvimento de Software. São Paulo, Brasil. Editora Bookman. 2014.