

MINISTÉRIO DA EDUCAÇÃO
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA
DIRETORIA DE ENSINO (DIREN)
DEPARTAMENTO DE ENSINO SUPERIOR (DEPES)
DEPARTAMENTO DE INFORMÁTICA (DEPIN)
CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET (CST-SI)

DEPARTAMENTO	PLANO DE CURSO DA DISCIPLINA
DEPIN - Departamento Acadêmico de Informática	ARQUITETURA E PADRÕES DE SOFTWARE

CÓDIGO	PERÍODO	ANO	SEMESTRE	PRÉ-REQUISITOS
GTSI 1252	5º	2012	2	GTSI1242 Programação de Software para WEB
CRÉDITOS	AULAS/SEMANA			TOTAL DE AULAS NO SEMESTRE
4	TEÓRICA	PRÁTICA	ESTÁGIO	72
	4	0	0	GTSI1246 Análise e Projeto de Sistemas

EMENTA
<p>Introdução aos conceitos de padrão de software e arquitetura de software; padrões para organizar a arquitetura de aplicações corporativas, padrões de projeto (<i>design patterns</i>): GoF; JEE; padrões da abordagem <i>Domain Driven Design</i> (DDD).</p>

BIBLIOGRAFIA
<p>Bibliografia básica</p> <ol style="list-style-type: none"> 1. EVANS, Eric, Domain-Driven Design Atacando As Complexidades na Criação do Software, Rio de Janeiro: Alta Books, 2009. ISBN: 9788576083603. 2. FOWLER, Martin. Padrões de arquitetura de aplicações corporativas. Porto Alegre: Bookman, 2006. xiii, 493 p., il. ISBN 9788536306384. 3. GAMMA, Erich, HELM, Richard, JOHNSON, Ralph & VLISSIDES, John, Padrões de Projeto – Soluções Reutilizáveis de Software Orientado a Objetos, 2ª edição, Porto Alegre: Bookman, 2000. ISBN 9788573076103. <p>Bibliografia complementar</p> <ol style="list-style-type: none"> 1. ALUR, Deepak; CRUPI, John; MALKS, Dan. Core J2 EE: as melhores práticas e estratégias de design. 2.ed.rev.atual. Rio de Janeiro: Campus, 2004. xxiv, 587p., il. ISBN 8535212728. 2. ELLIOTT, James e O'BRIE, Timothy M., Dominando Hibernate, Rio de Janeiro: Alta Books, 2009. ISBN: 9788576082446. 3. LARMAN, Craig. Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao processo unificado. 2.ed. Porto Alegre: Bookman, 2005. xiv, 607p., il., ISBN 8536303581. 4. FREEMAN, Eric et al. Use a cabeça: padrões e projetos. 2.ed. rev. Rio de Janeiro: Alta Books, 2009. xxiv, 478p., il. ISBN 9788576081746. 5. BLOCK, Joshua, Java Efetivo, 2a edição, Rio de Janeiro: Alta Books.

OBJETIVO GERAL

Familiarizar os alunos com conceitos avançados de desenvolvimento de software com o uso de padrões de software. Desenvolver a capacidade de identificar problemas em sistemas de software, analisá-los, e propor soluções baseadas no uso de arquiteturas e padrões de software.

METODOLOGIA

- Aulas expositivas com recursos audiovisuais
- Desenvolvimento de estudos de casos em laboratório, com uso de desenvolvimento dirigido por testes.
- Atendimento e orientação do professor visando o desenvolvimento do projeto de um sistema de software que envolva padrões estudados na disciplina.

CRITÉRIO DE AVALIAÇÃO

A avaliação semestral da envolve duas provas escritas (P1 e P2). As datas das provas estão no calendário do CEFET. Haverá também trabalhos individuais envolvendo os assuntos da disciplina. Os pesos de cada componente são os seguintes: as provas valem 80% da nota e os trabalhos valem os restantes 20%.

A média semestral (MS) será calculada pela fórmula $MS = [(0,8 \times P1 + 0,2 \times T1) + (0,8 \times P2 + 0,2 \times T2)]/2$

Na expressão acima, T1 e T2 são as médias dos trabalhos passados na primeira e segunda fase do curso, respectivamente.

O aluno que faltar a uma das duas provas (P1 ou P2) terá direito a uma avaliação alternativa, denominada segunda chamada, versando sobre todos os tópicos abordados no curso, e cuja data também é agendada entre docente e discentes. A nota obtida nessa 2ª chamada substituirá a da avaliação P1 ou P2 onde o aluno não esteve presente. Caso ele falte às duas avaliações, terá atribuído o grau ZERO em uma delas.

Segundo o regimento do CEFET-RJ, caso o aluno obtenha média parcial inferior a 3,0 (três e zero) estará reprovado diretamente. Graus MP maiores ou iguais a 7,0 (sete e zero) aprovam diretamente o aluno. Em situações onde o aluno tenha grau MP entre 3,0 inclusive e 7,0 exclusive, terá direito a uma prova final (PF), que, juntamente com a média parcial gerará uma nova média, denominada média final (MF). Essa média é calculada da seguinte forma:

$$MF = (MP + PF) / 2$$

Para ser aprovado, o aluno deve alcançar uma MF maior ou igual a 5,0 (cinco e zero). Caso contrário, estará reprovado, devendo repetir a componente curricular.

CHEFE DO DEPARTAMENTO

NOME	ASSINATURA

PROFESSOR RESPONSÁVEL PELA DISCIPLINA

NOME	ASSINATURA

PROGRAMA

1. Introdução aos conceitos de arquitetura e padrões de software
 - 1.1. Definição de padrão de software: vantagens e desvantagens, classificação; exemplos de aplicações.
 - 1.2. Princípios de projeto: acoplamento, coesão, generalização, composição, polimorfismo.
 - 1.3. Definição de arquitetura de software: SmartUI e Padrão Camadas (*layers pattern*)
2. Padrões de software para arquitetura de aplicações corporativas
 - 2.1. Organização da lógica da apresentação: Front Controller, MVC.
 - 2.2. Organização da lógica do domínio: Domain Model.
 - 2.3. Organização da lógica de acesso a dados: Active Record, Data Mapper, DAO. Uso do JPA para organização da camada de persistência.
3. Padrões de projeto
 - 3.1. Padrões GoF: visão geral, motivação, classificação
 - 3.2. Padrões GoF, parte I: Adapter, Aggregate, Singleton, Iterator, Template Method, Decorator
 - 3.3. Padrões GoF, parte II: Factory Method, Abstract Factory, Façade
 - 3.4. Padrões GoF, parte III: Strategy, State, Chain of Responsibility, Command
4. Domain Driven Design (DDD)
 - 4.1. Introdução aos conceitos de DDD
 - 4.2. Padrões DDD: Repositórios, Entidades, Agregados, ObjetosValor, Serviços do domínio.