

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)
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO (BCC)

DEPARTAMENTO		PLANO DE CURSO DA DISCIPLINA				
DEPIN - Departamento Acadêmico de Informática		PROGRAMAÇÃO DE SOFTWARE PARA WEB				
CÓDIGO	PERÍODO	ANO	SEMESTRE	PRÉ-REQUISITOS		
GCC 1415	4º	2012	2	GCC 1311 Programação Orientada a Objetos		
CRÉDITOS	AULAS/SEMANA					TOTAL DE HORAS NO SEMESTRE
4	TEÓRICA	PRÁTICA	ESTÁGIO			72
	4	0	0			

EMENTA

Configuração e Visão Geral do Ambiente WEB. A Tecnologia Servlet. A Tecnologia JavaServer Pages (JSP). Aplicações em Três Camadas na Plataforma Java. Acesso a banco de dados com JDBC. Gerenciamento de Sessão. TagLibs. Logging. Java Persistence API.

BIBLIOGRAFIA

Bibliografia básica

1. BASHAM, Bryan e SIERRA, Kathy, Use a Cabeça! Servlets & JSP, 2ª edição, Rio de Janeiro: Alta Books, 2009. ISBN: 9788576082941.
2. HALL, Marty e BROWN, Larry, Core Servlets e JavaServer Pages, Rio de Janeiro: Ciência Moderna, 2005. ISBN: 8573934328.
3. KURNIAWAN, Budi, Java para a Web com Servlets, JSP e EJB, Rio de Janeiro: Ciência Moderna, 2002. ISBN: 8573932104.

Bibliografia complementar

1. COAR, Ken e BOWEN, Rich, Apache Guia Prático, Rio de Janeiro: Alta Books, 2009. ISBN: 9788576082989.
2. MALKS, Dan, ALUR, Deepak e CRUPI, John, Core J2EE Patterns: As Melhores Práticas e Estratégias de Design, 2ª edição, Rio de Janeiro: Campus, 2004. ISBN: 8535212728.
3. BROGDEN, B. e MINNICK, C., Guia do Desenvolvedor Java – desenvolvendo e-commerce com Java, XML e JSP.
4. DIAS, Cláudia, Usabilidade na web: criando portais mais acessíveis, Rio de Janeiro: Alta Books, 2003.
5. ALUR, Deepak, CRUPI, John e MALKS, Dan, CORE J2EE Patterns – Melhores Práticas e Estratégias de

OBJETIVO GERAL

Capacitar o aluno em procedimentos e técnicas de desenvolvimento de aplicações para a WEB, com a utilização da plataforma Java. Desenvolver um estudo de caso para consolidar o conhecimento adquirido durante as aulas.

METODOLOGIA

- Aulas expositivas, eventualmente contando com recursos audiovisuais.
- Resolução de exercícios de fixação e propostos.

CRITÉRIO DE AVALIAÇÃO

A avaliação semestral envolve duas provas escritas (P1 e P2). As datas das provas são agendadas entre o professor e a turma. A média parcial (MP) será calculada pelo cômputo da média aritmética simples entre a nota P1 e P2:

$$MP = (P1 + P2) / 2$$

O aluno que faltar a uma das duas provas 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. Configuração e Visão Geral do Ambiente WEB

- 1.1. Instalação básica do ambiente - Instalando o Eclipse, TomCat e Plugins
- 1.2. Estrutura padrão para aplicação WEB
- 1.3. Estrutura do projeto – Explicando os diretórios
- 1.4. Adicionando pastas ao projeto
- 1.5. Criação de arquivos JSP
- 1.6. Escrevendo um JSP
- 1.7. Depurando projetos
- 1.8. O protocolo http
 - Solicitações HTTP
 - Respostas HTTP

2. A Tecnologia Servlet

- 2.1. Introdução
- 2.2. Como funciona um Servlet
- 2.3. Implementando Servlet HTTP
- 2.4. Falando um pouco sobre Container Servlet – O TomCat
- 2.5. Etapas para se escrever um Servlet
- 2.6. Construindo um Servlet simples
- 2.7. Servlets com mais detalhes
 - Ciclo de vida de um Servlet
 - O método init()
 - O método service()
 - O método destroy()
 - O Contexto de Servlet – O objeto “Servlet Context”
 - Como obter o Servlet Context
 - Principais métodos do objeto Servlet Context
 - Compartilhando informações entre Servlets – Uma visão Prática
 - Solicitações e Respostas (Request e Response)
 - A interface “Servlet Request”
 - A interface “Servlet Response”
- 2.8. Servlets e o protocolo http
 - A classe HttpServlet
 - As interfaces HttpServletRequest e HttpServletResponse
 - Obtendo parâmetros a partir de HttpServletRequest
 - Um pouco mais sobre HttpServletResponse
 - Solicitação de Despacho – Request Dispatcher

3. A Tecnologia JavaServer Pages (JSP)

- 3.1. Introdução

3.2. Qual o problema com Servlets?

3.3. Mecanismo de funcionamento de uma página JSP

3.4. Elementos de um JSP

- Comentários nos arquivos JSP
- Elementos de Diretivas
 - Diretiva Page
 - Diretiva Include
- Elementos de Script
 - Declarações
 - Scriptlets
 - Expressões
 - Elementos de ações

3.5. JavaBeans e JSP Beans

4. Aplicações em Três Camadas na Plataforma Java

4.1. Arquitetura Java para WEB

4.2. *Arquitetura modelo 1* versus *Arquitetura modelo 2*

4.3. Padrões Front Controller, Command e DAO

4.4. Testes em aplicações WEB

5. Acesso a banco de dados com JDBC

5.1. Inserindo dados no banco de dados

5.2. Deletando dados do banco de dados

5.3. Atualizando dados no banco de dados.

5.4. Pool de conexões

6. Gerenciamento de Sessão

6.1. Manipulando a sessão do usuário

6.2. Técnicas para Gerenciamento de Sessão

- Utilizando Campos Ocultos
- Utilizando Cookies
- Objetos Session
 - Mecanismo de identificação de um “Session Identifier”
 - A interface `javax.servlet.HttpSession`

7. TagLibs

7.1. Introdução

7.2. Instalação e Configuração

8. Logging

9. Java Persistence API (Hibernate): configuração e operações CRUD da API.