

PLANO DE APRENDIZAGEM

1. DADOS DE IDENTIFICAÇÃO			
Curso: Sistemas de Informação			
Disciplina: Programação Avançada		Código: SIF6N182	
Professor: Ronierison de Souza Maciel		Email: ronierison.maciел@unirios.edu.br	
CH Teórica: 50h	CH Prática: 30h	CH Total: 80h	Créditos: 04
Pré-requisito: Linguagem de Programação II			
Período: VI		Ano: 2021.1	

2. EMENTA:

Fundamentos Básicos de WEB: Aplicações para Internet/Intranet e Linguagem Java; Tipos de Aplicações: *Applets*, *Applications* e *Servlets*; Estrutura da Linguagem; Estrutura dos Pacotes; Aplicações Cliente/Servidor.

Desenvolvimento Web: *JavaServer Faces*; Interfaces gráficas em aplicações *WEB* (Primefaces); *Java Persistence API (JPA)*; *Hibernate*; Sessões; *Spring Security*.

3. COMPETÊNCIAS E HABILIDADES DA DISCIPLINA:

- Realizar a identificação de problemas que possuam solução algorítmica. As soluções se darão por meio da modelagem das classes do sistema e como este se comportará para resolver problemas reais.
- Resolver problemas usando ambientes de programação. Para alcance disto, será essencial o conhecimento em *JavaServer Faces* com a aplicação no projeto *web*.

4. OBJETIVO DA APRENDIZAGEM:

Desenvolver no aluno a Capacidade de desenvolver sistemas completos para internet/intranet, utilizando os conceitos mais avançados de tecnologia.

Capacitar o aluno a projetar aplicações cliente/servidor; desenvolver habilidade de programação com uma linguagem de programação *Web*; demonstrar metodologias de desenvolvimento rápido; capacitar o aluno a utilizar técnicas e ferramentas em alta no mercado de trabalho para *Web*.

5. CONTEÚDO PROGRAMÁTICO:

5.1. 1ª ETAPA:

5.1.1. CONTEÚDOS PRESENCIAIS (30 horas)

1. *EXtensible HyperText Markup Language (XHTML)* (03 h)
2. Aplicações para Internet/Intranet e Linguagem Java EE (03 h)
3. Tecnologias *Servlets* e JSP (03 h)
4. Classe Genérica (*Generics*) (03 h)
5. Framework MVC (03 h)
6. *JavaServer Faces (JSF)* (09 h)

- 6.1. Componentes JSF
- 6.2. Escopos JSF
- 6.3. Facelets
- 7. Java Persistence API (JPA) (03 h)
 - 7.1. Persistência de dados
 - 7.2. Diferenças entre modelos Relacional e OO
 - 7.3. Mapeamento Objeto-Relacional
- 8. Hibernate (03 h)

5.1.2. CONTEÚDOS NO AMBIENTE VIRTUAL DE APRENDIZAGEM (10 horas)

- 1. Atividade de transformação de um projeto HTML para XHTML (02 h)
- 2. Atividades sobre tecnologias que antecederam o JSF (02 h)
- 3. Definição da arquitetura MVC do projeto da disciplina (02 h)
- 4. Criação e configurações iniciais do Projeto no Eclipse IDE (02 h)
- 5. Modelagem Relacional do Banco para o projeto da disciplina (02h)

5.2. 2ª ETAPA:

5.2.1. CONTEÚDOS PRESENCIAIS (30 horas)

- 1. Filtros JSF (03h)
- 2. Biblioteca de Componentes (03h)
 - 2.1. Primefaces
- 3. Interfaces gráficas em aplicações WEB (09h)
 - 3.1. Criação de um template
 - 3.2. Uso de CSS com JSF
 - 3.3. Internacionalização do sistema
 - 3.4. Biblioteca Primefaces
- 4. Sessões (09h)
- 5. Spring Security (06h)
 - 5.1. Métodos de Autenticação e Autorização
 - 5.2. Configuração do Projeto com Spring Security
 - 5.3. Criação da página de login
 - 5.4. Controle de permissões de acesso

5.2.2. CONTEÚDOS NO AMBIENTE VIRTUAL DE APRENDIZAGEM (10 horas)

O quadro abaixo corresponde às fases de entrega do projeto. Será considerado a entregas incrementais, de modo que o professor consiga acompanhar e ver o desempenho no processo de desenvolvimento do *software* das equipes.

O cronograma com as datas de entrega ficará a cargo do professor e estará disponível para as equipes no início da 2ª etapa. Caso o aluno não entregue a atividade no prazo estabelecido, receberá nota zero, sem exceção. Na fase de entrega da apresentação, se o *software* não executar será atribuída a nota zero para o projeto independente das notas parciais. A atribuição das notas e atividades, pelo professor, seguirá o cronograma de fases como segue:

Fases	Atividade	Valor
Fase 1	Início do Projeto e distribuição dos temas	-
Fase 2	Criação do Projeto com seus pacotes MVC e entidades	0,5
Fase 3	Implementação das controladoras e DAOs	1,5
Fase 4	Criação das telas do sistema	1,0
Fase 5	Implementação das funcionalidades parciais do sistema	1,0

Fase 6	Apresentação do Projeto Completo + entrega do documento de <i>software</i>	6,0
Total		10,0

6. METODOLOGIA DO TRABALHO:

A disciplina utilizar-se-á de diversas mídias, tendo a prática como fio condutor do processo de aprendizagem a partir da pesquisa como princípio educativo. A disciplina terá conteúdos e atividades disponibilizadas em Ambiente Virtual de Aprendizagem (AVA-SAGAH), visando à sinergia entre as estratégias de inovação no uso de tecnologias digitais de informação e comunicação (TDIC) e os objetivos da disciplina, com vistas a promover aprendizagem significativa e colaborativa.

A aprendizagem dos conteúdos é baseada em metodologias que inserem o aluno em situações do seu cotidiano como profissional.

A **aprendizagem baseada em projetos**, a **aprendizagem por equipes** e a **instrução por pares** (peer instruction) por serem comprovadamente as formas mais eficazes de desenvolvimento de competências. Será adotado o modelo de **sala de aula invertida**, de acordo com o próprio ritmo do ambiente virtual e adaptação do acadêmico as metodologias e conteúdo, com resolução de problemas e execução de projetos. O conhecimento prima pela **aprendizagem adaptativa**, com o conteúdo trabalhado em uma plataforma de aprendizagem que permite a inserção de novos elementos de aprendizagem, e considerando os cenários de evolução individuais e coletivos. A solução personalizada otimiza o tempo de estudo do acadêmico e eleva seus índices de aprendizagem. Partindo do princípio da utilização da sala de aula invertida, auto estudo e resolução de problemas, a metodologia poderá ser utilizada tanto no PRESENCIAL CONECTADO quanto no PRESENCIAL, podendo ainda ser aplicada na modelagem híbrida, em que é considerada a mesclagem entre os dois modelos.

1ª ETAPA:

1) Aprendizagem Baseada em Problemas (ABP) (10,0 pontos)

Serão propostos problemas a serem resolvidos individualmente ou em duplas sobre um determinado conteúdo abordado em sala sobre Desenvolvimento *Web*, são eles: bibliotecas para *software web*; JPA; Hibernate; Primefaces; Sessões; e, *Spring Security*. O aluno poderá realizar estudos específicos e apresentar as soluções para os problemas indicados pelo professor.

2) Avaliação Institucional (10,0 pontos)

A avaliação escrita será composta por questões objetivas e subjetivas, versando sobre todos os temas discutidos na 1ª Etapa da disciplina. A avaliação será concebida no mesmo modelo estrutural de elaboração de questões utilizados nas provas do ENADE, suas regras e orientações estarão expostas no campo de instruções. A avaliação será individual e aplicada dentro do calendário acadêmico. Caso ocorram alterações no Calendário Acadêmico de 2020, esta data poderá ser alterada e devidamente comunicada aos discentes.

2ª ETAPA

3) Projeto Integrador (10,0 pontos)

Para a resolução do projeto final da disciplina serão propostos problemas e situações reais de como os alunos poderão criar o sistema, gerenciar relatórios e as regras de negócios do mesmo. Isso será fundamental para um aprendizado amplo e disseminado para a turma, uma vez que todos deverão expor as propostas de resolução identificados dos problemas.

4) Avaliação Processual (10,0 pontos)

A avaliação escrita será composta por questões subjetivas, versando os temas discutidos na 1ª Etapa da disciplina. A avaliação será individual, e aplicada uma semana antes da Avaliação Institucional. Caso ocorram alterações no Calendário Acadêmico de 2020, estada data poderá ser alterada, e devidamente comunicada aos estudantes.

5) Avaliação Institucional (10,0 pontos)

A avaliação escrita será composta por questões objetivas e subjetivas, versando sobre todos os temas discutidos na 1ª Etapa da disciplina. A avaliação será concebida no mesmo modelo estrutural de elaboração de questões utilizados nas provas do ENADE, suas regras e orientações estarão expostas no campo de instruções. A avaliação será individual e aplicada dentro do calendário acadêmico. Caso ocorram alterações no Calendário Acadêmico de 2020, estada data poderá ser alterada e devidamente comunicada aos discentes.

7. RECURSOS:

<input type="checkbox"/> Sala (comum)	<input type="checkbox"/> Sala Configuração Flexível	<input checked="" type="checkbox"/> Laboratório(s) - agendar
<input type="checkbox"/> Práticas em Campo	<input checked="" type="checkbox"/> Kit multimídia	<input type="checkbox"/> Outros (informar)

8. SISTEMA DE AVALIAÇÃO:

1ª ETAPA

- a) Aprendizagem baseada em problemas (10,0 pontos)
- b) Avaliação Institucional (10,0 pontos)

2ª ETAPA

- a) Projeto Integrador (10,0 pontos)
- b) Avaliação Processual (10,0 pontos)
- c) Avaliação Institucional (10,0 pontos)

Observações:

- **Média das Etapas:** será igual a média aritmética das avaliações nas respectivas etapas.
- **Detalhes das atividades no item 6. METODOLOGIA DO TRABALHO**

9. ATENDIMENTO EXTRA CLASSE:

Em caráter complementar, o professor oferece atendimento, através do endereço eletrônico:
ronierison.maciел@unirios.edu.br

10. BIBLIOGRAFIA BÁSICA:

FRASSON, Rosicléia. **Java Para Web Na Prática. Construindo aplicações utilizando JSF, JPA e Primefaces.** eBook Kindle. Amazon Servicos de Varejo do Brasil Ltda.

HORSTMANN, Cay; GEARY, David. **Core JavaServer Faces.** 3ª ed. Rio de Janeiro: Alta Books, 2012.

GONÇALVES, Edson. **Desenvolvendo Relatórios Profissionais com iReport para Netbeans IDE.** Ciência Moderna, 2009.

ARAÚJO, Everton Coimbra de. **Desenvolvimento para WEB com Java.** Florianópolis: Visual Books, 2010.

LUCKOW, Décio Heinzemann; MELO, Alexandre Altair. **Programação Java para a Web.** São Paulo: Novatec, 2010.

11. BIBLIOGRAFIA COMPLEMENTAR:

HALL, MARTY e BROWN, LARRY. "Core Servlets e JavaServer Pages. Volume 1: Tecnologias Core". Ciência Moderna, 2005.

HORSTMANN, C.S., CORNELL, G., "Core Java 2, Volume 1: Fundamentals". Sun Microsystems Press / Prentice Hall, 1999.

HORSTMANN, C.S., CORNELL, G., "Core Java 2, Volume 2: Advanced Features". Sun Microsystems Press / Prentice Hall, 1999.

12. LEITURA COMPLEMENTAR:

13. INFORMACÕES COMPLEMENTARES:

O aluno que faltar a prova escrita da 1ª chamada terá o direito de fazer uma prova de Segunda Chamada, cujo conteúdo abrangerá todo o programa. O aluno não terá direito de fazer Segunda Chamada para substituir as notas das listas de exercícios, atividades em sala ou do projeto.

A verificação do rendimento escolar compreende: frequência e aproveitamento nos estudos, os quais devem ser atingidos conjuntamente;

É exigida a frequência mínima (FS) 75% (setenta e cinco por cento) das atividades;

O aluno com frequência suficiente (FS), e média das notas de avaliações do semestre entre 3,0 (três) e 7,0(sete vírgula zero) terá direito à prova final no semestre;

Será atribuída nota 0 (zero) ao aluno que não comparecer às atividades ou não apresentar trabalhos no prazo estabelecido;

Todas as avaliações serão expressas em notas graduadas de 0 (zero) a 10 (dez);

A nota mínima de aprovação na disciplina é 5,0 (cinco vírgula zero). A AVALIAÇÃO FINAL, abrangerá todo o conteúdo do programa.

14. APROVAÇÃO:

Aprovado em ____/____/____

homologado em ____/____/____

COORDENADOR

GERÊNCIA ACADÊMICA

OBS: As datas das avaliações poderão sofrer alterações de acordo com o disciplinado pela secretaria acadêmica da UniRios.

APÊNDICE I

PROJETO DA DISCIPLINA PROGRAMAÇÃO AVANÇADA

1- INTRODUÇÃO

Com o avanço tecnológico, o número de empresas que adotam os sistemas *web* tem crescido consideravelmente, uma vez que esses sistemas têm contribuído para o melhor gerenciamento de informações, sendo de forma rápida e acessível. O sistema *web* é um *software* hospedado na internet onde qualquer usuário cadastrado pode acessá-lo através de um navegador. Duas de suas grandes vantagens são a acessibilidade, que permite acesso de qualquer local que tenha ponto de internet; e, a independência da plataforma de acesso, ou seja, não depende da configuração do seu computador.

Sendo assim, a disciplina de Programação Avançada (PA), do 6º período do curso de Sistemas de Informação do Centro Universitário do Rio São Francisco (UniRios), busca propagar conhecimento através de um projeto de desenvolvimento de *Software Web* em parceria com a disciplina de Laboratório de Banco de Dados, concebendo assim o projeto interdisciplinar. Na disciplina de PA o professor responsável tem a preocupação de guiar os alunos com o uso da linguagem de programação Java para *web* (JSF, acrônimo para *JavaServer Faces* (2018)) e os demais conhecimentos necessários para o sucesso da construção do programa.

2- PROFESSOR MENTOR

O professor e mentor responsável será Me. Ronierison de Souza Maciel, cujo far-se-á presente para guiar os alunos nas soluções dos problemas de *software* e tecnologias adotadas para o desenvolvimento dos programas.

3- EQUIPE DE ALUNOS

Para os projetos a serem desenvolvidos, poderão ser formadas equipes de até três alunos. Estes deverão estar regularmente matriculados na disciplina de PA, sem exceção.

4- PERÍODO

O período de duração do projeto é de aproximadamente 4 meses. O tempo é referente à carga horária da disciplina de PA, distribuída em média nessa quantidade de meses.

5- OBJETIVO

O objetivo principal deste projeto é a construção de um programa de *software web* desenvolvido em JSF, usando a IDE de desenvolvimento Eclipse (2020) para resolução de um problema. O projeto será construído baseado na proposta de Aprendizagem Baseada em Problema (ABP), logo os alunos terão a responsabilidade de resolver problemas do cotidiano detectados no parâmetro socioeconômico de Paulo Afonso e região.

6- JUSTIFICATIVA

A ABP representa um método de aprendizagem que tem por base a utilização de problemas como ponto de partida para a aquisição e integração de novos conhecimentos. Em essência, promove uma aprendizagem centrada no aluno, sendo os professores meros facilitadores do processo de produção do conhecimento. Nesse processo, os problemas são um estímulo para a aprendizagem e para o desenvolvimento das habilidades de resolução (Barrows, 1986 apud Dourado e Souza, 2015).

Durante o processo de construção serão propostos problemas e situações reais de como os alunos poderão criar relatórios, telas e gerenciar as regras de negócios do sistema. Isso será fundamental para um aprendizado amplo e disseminado para a turma, uma vez que todos deverão expor as propostas de resolução identificados dos problemas.

Acredita-se que ao término do projeto o aluno estará mais apto para resolver problemas através de *softwares*, além de compreender melhor a integração de todo o conhecimento adquirido até o atual período cursado, disseminando o conhecimento das disciplinas e preparando inicialmente o aluno para estar apto para o mercado de trabalho.

7- FASES E DISTRIBUIÇÃO DAS NOTAS

O quadro abaixo corresponde às fases de entrega do projeto. Será considerado a entregas incrementais, de modo que o professor e mentor consiga acompanhar e ver o desempenho no processo de desenvolvimento do *software* das equipes.

O cronograma com as datas de entrega ficará a cargo do professor e estará disponível para as equipes no decorrer das aulas da disciplina. Caso o aluno não entregue a atividade no prazo estabelecido, receberá nota zero, sem exceção. **Na fase de entrega da apresentação, se o *software* não executar será atribuída a nota zero para o projeto independente das notas parciais.**

A atribuição das notas e atividades, pelo professor, seguirá o cronograma de fases como segue:

Fases	Atividade	Valor
Fase 1	Início do Projeto e distribuição dos temas	-
Fase 2	Primeira versão do documento contendo os requisitos funcionais e não-funcionais	0,5
Fase 3	Modelo Conceitual e Lógico do Banco	1,0
Fase 4	Diagramas de Casos de Usos e Classes	0,5
Fase 5	Protótipos descartáveis do sistema	0,5
Fase 6	Configuração básica do sistema integrado com o banco de dados	1,0
Fase 7	Implementação da Arquitetura MVC	0,5
Fase 8	Implementação dos Cadastros e fachada	0,5
Fase 9	Implementação da GUI dos demais Casos de Uso	1,0
Fase 10	Implementação das Regras de Negócio e Tratamento de Exceções	0,5
Fase 11	Apresentação dos Projetos + Especificações Funcionais	4,0
Total		10,0

8- RECOMENDAÇÕES FINAIS

Na apresentação os alunos deverão entregar a documentação do *software* por meio de uma cópia impressas de acordo com o cronograma. A parte referente a implementação deverá ser enviada para o e-mail do professor da disciplina de PA pelo e-mail <ronierison.maciел@gmail.com>, não sendo necessário entregar cópias físicas do código.

REFERÊNCIAS

JAVASERVER FACES TECHNOLOGY. Linguagem de Programação *Web*, JavaServer Faces. Disponível em <<http://www.oracle.com/technetwork/java/javace/javaserverfaces-139869.html>>. Acesso em julho de 2018.

ECLIPSE IDE. Ambiente de Desenvolvimento Integrado Java. Disponível em <<https://www.eclipse.org>>. Acesso em julho de 2018.

SOUZA, S. C.; DOURADO, L. Aprendizagem Baseada Em Problemas (ABP): Um Método De Aprendizagem Inovador Para O Ensino Educativo. HOLOS, v5, 2015.