

CURSO DE GRADUAÇÃO: Informática Biomédica		
SÉRIE: 2ª série	SEMESTRE LETIVO DO ANO: 2017/1	(X) 1º SEMESTRE () 2º SEMESTRE () 1º e 2º SEMESTRES

PLANO DE ENSINO DA DISCIPLINA**BLOCO I – IDENTIFICAÇÃO**

Código da Disciplina:	(não preencher)
Nome da Disciplina:	Técnicas Avançadas de Projeto de Software
Dia(s) da semana	Segunda-feira (EAD)
Horário(s) de aula	8h às 11h

• Enfoque:(1) Obrigatória(2) Optativa Eletiva PDCI Concomitante com disciplina obrigatória:

Curso: _____

Série: _____

• Observação:**Número de Alunos por Disciplina:** **BLOCO II - CARGA HORÁRIA DA DISCIPLINA**

Teórica (horas):	<input type="text" value="4"/>
Prática (horas):	<input type="text" value="16"/>
Teórico-Prática (horas):	<input type="text" value="25"/>
Carga Horária Total (horas):	<input type="text" value="45"/>
EAD (horas):	<input type="text" value="41"/>
Unidade Administrativa: Departamento	<input type="text" value="DECESA"/>

BLOCO III – RESPONSABILIDADE DO DOCENTE*

Docente regente da disciplina:	CH Teórica	CH Prática	CH Teórico-Prática
Juliana Herbert	4	16	25
Docente (s) colaborador (es) na disciplina (aqueles que ministram, no mínimo, 20% da carga horária total da disciplina)	CH Teórica	CH Prática	CH Teórico-Prática
1.			
2.			
3.			
4.			
5.			
Docente (s) convidado (s) na Disciplina:	CH Teórica	CH Prática	CH Teórico-Prática
1.			
2.			

* Docentes efetivos e substitutos vinculados à UFCSPA

BLOCO IV – DESCRIÇÃO DO PLANO DE ENSINO

Ementa: Apresenta e discute técnicas avançadas de desenvolvimento de software, com foco nas atividades das fases de projeto e codificação, através da apresentação de boas práticas e de ferramentas que as suportem.

Objetivo Geral: Os alunos devem compreender as técnicas de projeto e codificação de software apresentadas na disciplina e estar aptos para saber utilizá-las em contextos reais, com o auxílio de ferramentas, em sua prática profissional.

Objetivos Específicos: Para que os alunos sejam capazes de alcançar o objetivo geral, é necessário dominar os seguintes objetivos específicos:

- Compreender conceitos de código limpo e de padrões de código e saber aplicá-los em projetos de desenvolvimento de software;
- Conhecer boas práticas de codificação e sua aplicabilidade em diferentes contextos;
- Saber definir o conceito de gerência de configuração, incluindo processos relacionados e atividades, no contexto de desenvolvimento ágil de projetos de software;
- Conhecer *Test Driven Development* (TDD) e entender como utilizá-lo em projetos de software;
- Conhecer os *Design Patterns* e saber aplicá-los em projetos de software;
- Compreender e reconhecer os tipos de padrões a serem utilizados em projetos de software.

Conteúdo Programático:

1. Código Limpo
 - a. Conceito
 - b. Princípios
 - c. Escolas de Pensamento
 - d. Código Limpo no Contexto Ágil

2. Padrões de Código
 - a. Funções
 - b. Comentários
 - c. Formatação
 - d. Orientação a Objetos
 - e. Padronização de Código no Contexto Ágil

3. Gerência de Configuração
 - a. Conceitos
 - b. Atividades
 - c. Automação e Ferramentas
 - d. Gerência de Configuração no Contexto Ágil

4. *Test Driven Development* (TDD)
 - a. Conceitos
 - b. Exemplos
 - c. Padrões de TDD

5. *Design Patterns*
 - a. Conceitos
 - b. Exemplos
 - c. Catálogo de *Design Patterns*
 - d. Utilização de *Design Patterns*

Procedimentos Didáticos:

As aulas serão realizadas a distância, utilizando para tal o ambiente virtual de aprendizagem *Moodle*. Neste ambiente, serão utilizados os seguintes recursos: agenda, publicação de conteúdo, fóruns de discussão, questionários, envio de trabalhos, *feedback* e acompanhamento do acesso de cada aluno, entre outros. Também deverão ser utilizados textos no formato PDF e HTML, vídeos no formato MP4 e ferramentas *freeware* e preferencialmente *Open Source*, para a execução de exercícios práticos.

A parte conceitual da disciplina será apresentada através da indicação de textos e de páginas web relevantes e através da visualização de vídeos, elaborados para este objetivo.

Todo o seu conteúdo será trabalhado a partir de projetos de software desenvolvidos pelos alunos. Os projetos poderão estar associados a outras disciplinas obrigatórias, ou poderão ser definidos a partir da motivação pessoal dos próprios alunos, desde que contemplem os critérios apresentados no início da disciplina.

Situações e Critérios de Avaliação:

Os alunos serão avaliados pela entrega das atividades propostas. Os trabalhos estão organizados em duas categorias:

- Trabalhos ao longo do semestre, incluindo participação de fóruns e pequenos experimentos práticos dos conceitos apresentados – com o peso 4;
- Trabalho final – atividade presencial:
 - entrega, de peso 2;
 - apresentação, de peso 4.

Bibliografia Básica:

LARMAN, G. Utilizando UML e Padrões: uma Introdução à Análise e ao Projeto Orientado a Objetos. Bookman, Porto Alegre, 2007.

PRESSMAN, R.S., Engenharia de Software, 7a edição, Mc Graw Hill, 2010.

WAZLAWICK, R. S. Engenharia de Software: Conceitos e Práticas, 1a edição, Elsevier Editora, 2013.

Bibliografia Complementar:

Website *Software Engineering Information Repository*. Disponível em: <http://seir.sei.cmu.edu/seir/Welcome/>

Website da *Scrum Alliance*. Disponível em: <https://www.scrumalliance.org/>

Outras Fontes:

Normas de qualidade e de processos de software na área de Informática Biomédica.

Websites especializados.

BLOCO V – CRONOGRAMA
Cronograma de Atividades para **DISCIPLINAS**

Data (dd/mm)	Dia da semana	Horário	Turma	Conteúdo da aula	Aula*	EaD**	Laboratório	Professor que ministra a aula e/ou Professor/Palestrante convidado
20/2	2ª feira	8h às 11h	U	Atividades de ambientação e de nivelamento. Apresentação da disciplina. Publicação da apresentação pessoal dos alunos e apresentação de expectativas com o curso. Vídeo: Código limpo – conceitos. Fórum: impacto de práticas de código limpo no contexto ágil de desenvolvimento de software. Abertura do fórum de dúvidas.	TP	X		Juliana Herbert
27/2	2ª feira	8h às 11h	U	Princípios e escolas de pensamento do código limpo. Vídeo: Princípios do código limpo. Vídeo: Escolas de pensamento e código limpo no contexto ágil. Tarefa: análise de código próprio, na qual um aluno analisa um trecho de código que ele mesmo produziu,	TP	X		Juliana Herbert

				com base nos conceitos apresentados. Texto: <i>Links</i> para leituras complementares (não obrigatórias, mas úteis para nivelamento de conceitos).				
6/3	2ª feira	8h às 11h	U	Padrões de código. Páginas: geradas com infográficos sobre o assunto. Tarefa: análise de código por pares, na qual um aluno analisa um trecho de código de um colega, com base nos conceitos apresentados.	TP	X		Juliana Herbert
13/3	2ª feira	8h às 11h	U	Gerência de configuração. Vídeo: Conceitos gerais. Tarefa: construção colaborativa de <i>wiki</i> sobre a ferramenta <i>GitHub</i> . Tarefa: definição de projeto para o trabalho final da disciplina, a partir de lista de critérios apresentada.	TP	X		Juliana Herbert
20/3	2ª feira	8h às 11h	U	Gerência de configuração. Vídeo: Atividades. Tarefa: complementação colaborativa de <i>wiki</i> sobre a ferramenta <i>GitHub</i> , relacionando as características da ferramenta com as atividades apresentadas.	TP	X		Juliana Herbert
27/3	2ª feira	8h às 11h	U	Gerência de configuração. Tarefa: utilização do <i>GitHub</i> para a gerência de colaboração de membros de uma equipe, utilizando conceitos de gerência de configuração.	P	X		Juliana Herbert
3/4	2ª feira	8h às 11h	U	TDD (<i>Test Driven Development</i>). Vídeo: conceitos. Tarefa: implementação de TDD em exemplo simples de programa.	P	X		Juliana Herbert
10/4	2ª feira	8h às 11h	U	TDD (<i>Test Driven Development</i>). Tarefa: análise da implementação de TDD em tarefa da semana anterior.	TP	X		Juliana Herbert
17/4	2ª feira	8h às 11h	U	TDD (<i>Test Driven Development</i>). Vídeo: padrões de TDD.	TP	X		Juliana Herbert
24/4	2ª feira	8h às 11h	U	TDD (<i>Test Driven Development</i>). Tarefa: aplicação de padrões de TDD no projeto da disciplina.	P	X		Juliana Herbert
1/5	2ª feira	8h às 11h	U	<i>Design Patterns</i> . Vídeo: conceitos.	TP	X		Juliana Herbert

				Tarefa: leitura sobre um dos <i>Design Patterns</i> apresentados.				
8/5	2ª feira	8h às 11h	U	<i>Design Patterns</i> . Vídeo: Exemplos.	TP	X		Juliana Herbert
15/5	2ª feira	8h às 11h	U	<i>Design Patterns</i> . Vídeo: conceitos. Tarefa: apresentação de <i>Design Pattern</i> escolhido.	TP	X		Juliana Herbert
22/5	2ª feira	8h às 11h	U	<i>Design Patterns</i> . Vídeo: catálogo de <i>Design Patterns</i> . Tarefa: preparação do trabalho final, a partir de orientações publicadas no Moodle.	P	X		Juliana Herbert
29/5	2ª feira	8h às 11h	U	<i>Design Patterns</i> . Vídeo: utilização de <i>Design Patterns</i> . Tarefa: preparação do trabalho final, a partir de orientações publicadas no Moodle.	P	X		Juliana Herbert
5/6	2ª feira	8h às 11h	U	Apresentação dos trabalhos finais da disciplina.	T		Sala 203 – Prédio 3	Juliana Herbert
12/6	2ª feira	8h às 11h	U	Exame	T		Sala 203 – Prédio 3	Juliana Herbert

*Indicar como a aula será desenvolvida: Teórica (T), Prática (P) e/ou Teórico-Prática (TP)

**Assinalar se a aula será desenvolvida na modalidade de Educação a Distância (EaD)

Data de emissão: 15/12/2016

Professor Regente: Juliana Herbert

Chefe do Departamento: Cláudia Libânio

Coordenador do Curso: Cecília Flores