



DISCIPLINA: <b>TÓPICOS ESPECIAIS EM ENGENHARIA DE SOFTWARE - Qualidade de Produto de Software</b>	CÓDIGO: GT00ESO006.1
---	-------------------------

VALIDADE: Início: **Janeiro/2020**

Eixo: **Engenharia de Software**

Carga Horária: Total: **50 horas (60 horas-aula)** Semanal: **4 aulas** Créditos: **4**

Modalidade: **Teórica** Integralização: **Optativa**

Classificação do Conteúdo pelas DCN: **Profissional**

**Ementa:**

Fatores de qualidade de software. Modularidade de software. Coesão, acoplamento e conectividade. Princípios de projeto orientado a objetos. *Code smells. Refactoring.* Medição de produtos de software. Ferramentas de avaliação de qualidade de produto de software.

Curso(s)	Período
Engenharia de Computação	7º

Departamento/Coordenação: **Departamento de Computação - DECOM**

**INTERDISCIPLINARIDADES**

<b>Pré-requisitos</b>
Modelagem e Desenvolvimento de Software Laboratório de Modelagem e Desenvolvimento de Software
<b>Co-requisitos</b>
--
<b>Disciplinas para as quais é pré-requisito / co-requisito</b>
--
<b>Outras inter-relações desejáveis</b>
--

<b>Objetivos:</b> <i>A disciplina deverá possibilitar ao estudante</i>	
1	Conhecer, compreender e aplicar os conceitos e técnicas essenciais e atuais para a construção de software com alto padrão de qualidade interna.
2	Avaliar a qualidade interna de softwares e definir intervenções para a melhoria da qualidade interna do software.



Unidades de ensino		Carga-horária (horas-aula)
1	INTRODUÇÃO 1. Conceito de produto de software 2. Conceito de qualidade de produto de software 3. Dimensões de qualidade de produto de software	4
2	MODULARIDADE DE SOFTWARE 1. Conceito de módulo 2. Modularidade 3. Coesão 4. Acoplamento 5. Conectividade	8
3	PRINCÍPIOS DE CONSTRUÇÃO DE SOFTWARE MODULAR 1. Princípios para atingir modularidade a. Unidade linguística b. Baixa conectividade c. Interface pequena d. Interface explícita e. Ocultação de informação 2. Princípios de projeto orientado a objetos a. Preferência à composição b. Restrição de acesso c. Programação para interface d. SOLID	12
4	REFATORAÇÃO DE SOFTWARE 1. Code smells 2. Técnicas de refatoração 3. Estado da arte em ferramentas de detecção de <i>code smells</i>	10
5	MEDIÇÃO DE SOFTWARE 4. Tipos de métricas 5. Categorias de métricas 6. Métricas de software clássicas 7. Métricas de software orientado a objetos 8. Ferramentas de medição de software	16
<b>Total</b>		50 horas

### Bibliografia Básica

1	Meyer, Bertrand. <i>Object-Oriented Software Construction</i> . Second Edition. Prentice Hall. 1997.
---	--



2	Fowler, Martin. <i>Refatoração: Aperfeiçoando o Design de Códigos Existentes</i> . 2. ed. Novatec, 2020.
3	Pressman, Roger S. <i>Engenharia de software: uma abordagem profissional</i> . 8. ed. Porto Alegre: AMGH, 2016.

### Bibliografia Complementar

1	Wazlawick, Raul S.. <i>Engenharia de Software - Conceitos e Práticas</i> . Elsevier. 2013.
2	O'Regan, Gerard. <i>Introduction to Software Quality</i> . Springer. 2014.
3	Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John. <i>Padrões de Projetos. Soluções Reutilizáveis de Software Orientados a Objetos</i> . Bookman. 1. ed. 2000
4	<i>Marco Tulio Valente. Engenharia de Software Moderna: Princípios e Práticas para Desenvolvimento de Software com Produtividade, 2020.</i>
5	Fenton, Norman; Bieman, James. <i>Software Metrics</i> . 3rd Edition. CRC Press, 2014.
6	Pressman, Roger S. <i>Engenharia de software: uma abordagem profissional</i> . 8. ed. Porto Alegre: AMGH, 2016.
7	Artigos científicos sobre o conteúdo da disciplina.