

Professor: Estudos Autônomos - Sem Professor	Disciplina / *Unidade Curricular: Modelos, métodos e técnicas da engenharia de software Carga Horária: 160	Curso: Ciência da Computação Período Letivo: 2024/1 <i>* Exclusivo para Unidades Curriculares</i>
Objetivo Geral / *Tópico Gerador: Como implementar software através de conceitos, métodos e práticas da engenharia de software?; Quais os principais modelos de engenharia de software e como podem ser aplicados?; Modelos de processos de software baseados em cascata, prototipação, incremental e espiral; Estudo da viabilidade de software através da engenharia de requisitos; Como garantir o desenvolvimento de software por meio de uma arquitetura adequada e utilizando metodologias de projetos?; Qual o diferencial de uma solução que utiliza padrões de projetos?; Projetar e desenvolver softwares utilizando metodologias ágeis de desenvolvimento; Conceito de DevOps para integração e entrega contínua de software.		
Objetivos Específicos / *Metas de compreensão: Reconhecer as fases da engenharia de requisitos, bem como sua aplicação e importância desde o levantamento de requisitos até a gestão de requisitos; Aplicar princípios e práticas do desenvolvimento ágil em desenvolvimento de software; Identificar e avaliar a aplicabilidade dos modelos de processo de software e as boas práticas da engenharia de software; Diferenciar tipos de arquitetura de software, compreendendo seus benefícios e quais aplicá-los em determinadas situações; Aplicar integração e entrega contínua envolvendo práticas de DevOps; Analisar as diferentes técnicas para coleta de requisitos, e diferenciar como cada uma delas pode ser utilizada; Criar soluções com os principais padrões de projeto e boas práticas de arquitetura de software; Meta máxima: aplicar o desenvolvimento ágil com frameworks atuais, de acordo com seus artefatos;		
Plano Aula / Atividade Discente / *Desempenho de Compreensão: Os desempenhos de compreensão são constituídos por experiências, estudos, pesquisas e práticas desenvolvidos no decorrer do semestre letivo que possibilitam ao aluno expressar, de diferentes formas, evidências de que atingiu a compreensão. Tais desempenhos demonstram com clareza que os alunos dominam as metas de compreensão, por meio de sua atuação em projetos desafiadores e acessíveis que promovem o seu envolvimento reflexivo sobre a situação a ele exposta. Nesta Unidade Curricular, os alunos demonstrarão a sua compreensão por meio do(s) seguinte(s) desempenho(s): 4(descrição dos professores da UC do desempenho ou desempenhos definidos no planejamento).		
Avaliação Contínua: Atividade Avaliativa 1 (A1) 30pts Avaliação Integrada (AIT) 30pts (Extra) Atividade Avaliativa 3 (A3) 40pts Atividade Avaliativa 2 (A2) 30pts Exigência mínima para aprovação: 70 pontos e % de frequência nas disciplinas presenciais. Tipo de Avaliação: Avaliação de EAD		

Ementa:

Fundamentos e práticas de engenharia de software, capacitando os estudantes a compreenderem, diferenciarem e aplicarem os principais modelos de processos de desenvolvimento de software na criação de soluções computacionais. Paradigmas e frameworks de desenvolvimento ágil. Engenharia de requisitos compreendendo as técnicas de elicitación, análise, especificação, gestão, validação e negociação de requisitos, além de estudo de viabilidade de soluções computacionais. Compreensão dos tipos de arquitetura e padrões projeto de software e seu impacto sobre a modularidade, manutenibilidade e otimização dos recursos de uma solução computacional. Prototipação de interfaces com o usuário como técnica para a construção de sistemas eficientes e usáveis. Fundamentos de DevOps que integram desenvolvimento e operações para otimizar a entrega contínua de software.

Metodologia (Não se aplica a Unidades Curriculares, exceto para o curso de Medicina):

Programa:

Conceitos, métodos e práticas da engenharia de software. Modelos de processos de software. Modelo de processo cascata. Modelo de processo de prototipação. Modelo de processo incremental. Modelo de processo espiral. Integração e entrega contínua. Paradigma de desenvolvimento ágil. Manifesto ágil. Frameworks para desenvolvimento ágil de software. Visão geral DevOps.

***Certificação:**

Qualificação Profissional em Modelos, Métodos e Técnicas da Engenharia de Software

***Competências:**

Compreender e aplicar os diferentes modelos de processos de desenvolvimento de software para diferentes problemas computacionais; Identificar e analisar requisitos e especificações para problemas específicos e planejar estratégias para suas soluções, bem como sua gestão; Projetar e criar soluções computacionais com qualidade em harmonia com o ambiente social e físico no seu entorno de aplicação considerando os princípios e boas práticas de engenharia de software; Avaliar e tomar decisões com base no conhecimento de arquitetura de software e considerando aspectos de infraestrutura, consciente dos aspectos éticos, legais e dos impactos ambientais decorrentes; Aplicar os princípios de interação humano computador para avaliar e construir sistemas de software com enfoque na experiência do usuário; Compreender e aplicar práticas e princípios de desenvolvimento ágil para sistemas de software que se adequam a esse novo paradigma;

Bibliografia Básica:

MUNIZ, Antonio; SANTOS, Rodrigo; IRIGOYEN, Analia; MOUTINHO, Rodrigo. Jornada DevOps Unindo cultura ágil, Lean e tecnologia para entrega de software com qualidade. 1a. Edição. São Paulo: Editora Brasport, 2019.
SOMMERVILLE, Ian. Engenharia de Software. 9. ed. São Paulo: Pearson Prentice Hall, 2011. [https://bv4.digitalpages.com.br/?term=engenharia%2520de%2520software&searchpage=1&filtro=todos&from=busca&page=_14\\$ion=0#/legacy/276](https://bv4.digitalpages.com.br/?term=engenharia%2520de%2520software&searchpage=1&filtro=todos&from=busca&page=_14$ion=0#/legacy/276)
PRESSMAN, Roger; MAXIM, Bruce. Engenharia de Software. Uma abordagem profissional. 8a. Ed. Bookman, 2016. <https://integrada.minhabiblioteca.com.br/#books/9788580555349/cfi/3//4/2@100:0.00>

Emitido por: DBADM

Rio de Janeiro, 02/06/2025

Bibliografia Complementar:

MORAIS, Izabelly Soares de. Engenharia de software. São Paulo: Pearson, 2017. E-book. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/184098/pdf/0>.

RANGEL, Pablo. Sistemas orientados a objetos. Rio de Janeiro: Brasport, 2021. E-book. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/197367/epub/0>.

FOGGETTI, Cristiano. Gestão Ágil de Projetos. Pearson, 2015. <https://bv4.digitalpages.com.br/?term=desenvolvimento%2520%25C3%25A1gil&searchpage=1&filtro=todos&from=busca&page=-7§ion=0#/legacy/22131>

FLOWLER, Martin; SCOTT, Kendall. UML essencial: um breve guia para a linguagem-padrão de modelagem de objetos. 3ª. ed. Porto Alegre: Bookman, 2004. <https://integrada.minhabiblioteca.com.br/#/books/9788560031382/cfi/6/2/1/4/2@0:0.131>

PFLIEGER, Shari Lawrence. Engenharia de software: teoria e prática. 2. ed. São Paulo: Prentice Hall, 2004. <https://bv4.digitalpages.com.br/?term=engenharia%2520de%2520software&searchpage=1&filtro=todos&from=busca#/legacy/476>