

Plano de Ensino

Curso			Semestre/Ano
Tecnologia em Análise e Desenvolvimento de Sistemas			2o Semestre/2025
Disciplina			Sigla
Engenharia de Software II			IES200
Carga Horária Semanal	Carga Teórica	Carga Prática	Carga Horária Semestral
4	2	2	80
Professor			
DEIVISON SHINDI TAKATU			
Ementa			
A disciplina apresenta os fundamentos conceituais da Engenharia de Software e da Engenharia de Requisitos, abordando seus conceitos, tipos, elicitação, análise, negociação e especificação. Introduz a Linguagem de Modelagem Unificada (UML), explorando diagramas estruturais (classes, objetos, pacotes, componentes e implantação) e comportamentais (casos de uso, atividades, sequência, comunicação e máquina de estados). Enfatiza ainda a documentação e a padronização de projetos segundo normas da ABNT, a organização de portfólio digital e a integração do projeto final, promovendo a aplicação prática dos conceitos estudados.			
Objetivo			
Desenvolver competências em requisitos e modelagem UML, aplicando normas e documentação para criar projetos de software organizados, claros e eficientes.			
Metodologia			
Aulas expositivas dialogadas, estudos de caso, exercícios práticos, uso de UML, trabalho em equipe, portfólio digital e projetos integradores.			
Critérios de Avaliação			
Fórmula : $(P1 * 0.25) + (P2 * 0.25) + ((PJ + AT) * 0.50)$			
Legendas :			
P1 - Prova 1			
P2 - Prova 2			
PJ - Projeto			
AT - Atividades			

Plano de Ensino

Plano de Aula

- 1 Apresentação do Plano de Ensino -> Apresentação do Plano de Ensino
- 2 Introdução à disciplina e revisão de engenharia de software – Fundamentos conceituais
- 3 Apresentação da disciplina, ementa, objetivos e critérios de avaliação. Revisão dos principais conceitos de Engenharia de Software.
- 4 Dinâmica de apresentação dos alunos e levantamento de expectativas. Exercício rápido de revisão sobre fundamentos da disciplina.
- 5 Engenharia de requisitos – Conceitos e tipos de requisitos funcionais e não funcionais
- 6 Estudo dos conceitos básicos de requisitos, importância e classificação (funcionais e não funcionais).
- 7 Análise de pequenos casos para identificação de requisitos funcionais e não funcionais.
- 8 Engenharia de requisitos – Elicitação, análise, negociação e especificação de requisitos
- 9 Discussão sobre técnicas de elicitação (entrevistas, questionários, observação). Etapas de análise, negociação e especificação.
- 10 Simulação de entrevista com “cliente” fictício e construção inicial de uma lista de requisitos.
- 11 Linguagem de modelagem unificada (UML) – Visão geral e princípios básicos
- 12 Introdução à UML, histórico, objetivos e principais elementos de modelagem.
- 13 Exercício prático de interpretação de diagramas UML simples.
- 14 Diagramas estruturais – Classes, objetos e perfil
- 15 Conceitos, notação e finalidade dos diagramas de classes e objetos. Perfis de modelagem.
- 16 Criação de um diagrama de classes a partir de um estudo de caso fornecido.
- 17 Diagramas estruturais – Pacotes, componentes e implantação
- 18 Definição e uso de diagramas de pacotes, componentes e implantação.
- 19 Exercício de modelagem de pacotes e de componentes para um sistema de exemplo.
- 20 Diagramas comportamentais – Casos de uso e atividades
- 21 Apresentação dos diagramas de casos de uso e atividades. Notação e interpretação.
- 22 Elaboração de um diagrama de casos de uso e de atividades para um cenário proposto.
- 23 Diagramas comportamentais – Sequência, comunicação e máquina de estados
- 24 Estudo dos diagramas de sequência, comunicação e máquina de estados.
- 25 Construção de diagramas de sequência e comunicação a partir de um caso prático.
- 26 Documentação e padronização de projetos – Requisitos e diagramas segundo normas ABNT
- 27 Importância da padronização na documentação de software. Principais normas da ABNT aplicáveis.
- 28 Análise de exemplos de documentos técnicos em conformidade com normas.
- 29 Documentação e padronização de projetos – Organização e portfólio digital
- 30 Estruturação de documentação de projetos em formato digital. Organização de portfólio acadêmico/profissional.
- 31 Início da elaboração do portfólio digital do projeto final.
- 32 Atividade de documentação de projetos – consolidação do material final
- 33 Orientação para finalização da documentação de projetos.
- 34 Entrega e revisão coletiva da documentação consolidada.
- 35 Revisão preparatória – avaliação substitutiva
- 36 Revisão geral e esclarecimento de dúvidas.
- 37 Resolução de exercícios simulados e estudos de caso.
- 38 Entrega das médias finais e plantão de dúvidas
- 39 Apresentação dos resultados finais e feedback individual/coletivo.
- 40 Avaliação final dos projetos

Bibliografia Básica

PRESSMAN, Roger S.; MAXIM, Bruce R. Engenharia de Software: uma abordagem profissional. 8. ed. McGraw-Hill, 2016.

SOMMERVILLE, Ian. Engenharia de Software. 10. ed. Pearson, 2019.

LARMAN, Craig. Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos. 3. ed. Bookman, 2007.

Plano de Ensino

Bibliografia Complementar

PFLEEGER, Shari Lawrence; ATLEE, Joanne M. Engenharia de Software: teoria e prática. 4. ed. Pearson, 2010.

JACOBSON, Ivar; BOOCH, Grady; RUMBAUGH, James. UML: Guia do Usuário. 2. ed. Bookman, 2000.

KOTONYA, Gerald; SOMMERVILLE, Ian. Engenharia de Requisitos. 1. ed. LTC, 1998.

Bibliografia Referência

IEEE Computer Society. Guide to the Software Engineering Body of Knowledge (SWEBOK V3.0). IEEE, 2014.

ISO/IEC/IEEE. 29148:2018 – Systems and software engineering – Life cycle processes – Requirements engineering. ISO/IEC/IEEE, 2018.

Object Management Group (OMG). Unified Modeling Language (UML) Specification. Versão 2.5.1, 2017.