



UNIVERSIDADE FEDERAL DE OURO PRETO
PRÓ-REITORIA DE GRADUAÇÃO
PLANO DE ENSINO



Nome do Componente Curricular em português: Engenharia de Software II		Código: BCC323
Nome do Componente Curricular em inglês: Software Engineering II		
Nome e sigla do departamento: Departamento de Computação (DECOM)		Unidade acadêmica: ICEB
Nome dos docentes: Johnatan Alves de Oliveira		
Carga horária semestral 60 horas	Carga horária semanal teórica 04 horas/aula	Carga horária semanal prática 0 horas/aula
Data de aprovação na assembleia departamental:		
Ementa: Processo de Desenvolvimento de Software. Engenharia de Requisitos. Verificação, Validação e Teste de Software. Medição. Gerência de Configuração de Software. Evolução de Software. Gerência de projetos de Software. Qualidade de Software.		
Conteúdo programático: 1) Processo de Desenvolvimento de Software <ul style="list-style-type: none">▪ Processos, modelos de processos, ciclo de vida de software.▪ Ciclo em cascata▪ Espiral▪ Processo unificado▪ Métodos ágeis▪ Gerência de projetos 2) Engenharia de Requisitos <ul style="list-style-type: none">▪ Elicitação de requisitos▪ Requisitos funcionais, não-funcionais e regras de negócio▪ Técnicas de levantamento de requisitos (Questionário, Entrevistas, entre outras)▪ Documentação de requisitos e casos de uso▪ Ferramenta de suporte a análise de requisitos 3) Qualidade e métricas <ul style="list-style-type: none">▪ Controle e garantia de qualidade▪ Modelo de maturidade: CMMI e MPSBR▪ Princípios de medição▪ Medições de software orientado a objetos		

- Métricas de código fonte
- Métricas para teste
- Métricas de manutenção
- Aplicações

4) Verificação e Validação de Software

- Distinguir entre verificação e validação
- Diferentes tipos de testes: interface humano-computador, usabilidade, confiabilidade, segurança, conformidade com a especificação
- Conceitos básicos: criação de plano de teste, geração de casos de teste caixa-preta e caixa-branca e Test-Driven Development

Objetivos:

Dar ao aluno condições de perceber o desenvolvimento de software como um processo de engenharia, baseado em planejamento, medição e melhoria contínua.

Metodologia:

- Aulas teóricas: aulas expositivas onde poderão ser usadas ferramentas síncronas (webconferências) e assíncronas no ambiente virtual de aprendizagem.
- Aulas práticas: webconferências (Google Meet) para resolução de exercícios e estudos dirigidos com atividades individuais (ou em grupos) contendo exercícios e/ou implementações dos métodos estudados.
- Preparação de videoaulas e outros tipos de materiais complementares.
- Leituras recomendadas de textos técnicos, apostilas e livros de autores especialistas com a finalidade de proporcionar ao discente a oportunidade de consulta e desenvolvimento de sua capacidade de análise, síntese e crítica de uma bibliografia específica.

Recursos utilizados:

Para o desenvolvimento da aprendizagem serão adotados, concomitantemente e em todos os tópicos da disciplina, os seguintes recursos de apoio didático-pedagógico:

- Computador desktop ou notebook.
- Acesso à internet para pesquisa de materiais de interesse da disciplina e execução das atividades.
- Ferramentas do ambiente virtual de aprendizagem adotado (Moodle e Google Forms).
- Videoaulas e webconferência (via Google Meet) com apresentação multimídia para as aulas expositivas dialogadas.

Atividades avaliativas:

Dois provas teóricas (P1 e P2), 5 listas de exercícios (L1, L2, L3, L4 e L5) e um trabalho prático (TP), todos avaliados em 10 pontos.

$$Média\ Final = 0.25P1 + 0.25P2 + 0.4TP + 0.02L1 + 0.02L2 + 0.02L3 + 0.02L4 + 0.02L5$$

Datas das avaliações:

Prova 1: 26/10/2021

Prova 2: 14/12/2021

Exame Especial: 11/01/2022 O Exame Especial será uma prova única, receberão uma prova com questões referentes aos conteúdos estudados durante todo o semestre e terão um tempo determinado para submeter a folha de respostas na plataforma.

Observações:

- Os procedimentos e o cronograma das avaliações poderão ser alterados a qualquer momento ao longo do semestre a partir de discussões coletivas com os estudantes.
- Trabalho Prático: os alunos deverão gerar um relatório com todas as etapas de desenvolvimento de software, desde o levantamento de requisitos, testes e entrega do software. Para isso, os alunos irão codificar em linguagem de programação livre. Sendo assim, não há limitação de quais linguagens e tecnologias poderão ser utilizadas.
- Provas: Os alunos receberão uma prova com questões referentes aos conteúdos estudados e terão um tempo determinado para submeter a folha de respostas na plataforma indicada pelo professor.
- Frequência: será computada mediante a entrega das atividades, considerando também as contribuições nas discussões, participação dos fóruns e colaboração para boa conduta da disciplina.
- Para ser aprovado na disciplina é necessária uma nota mínima de 6,0 (seis) pontos em uma escala de zero a dez e frequência mínima de setenta e cinco por cento.

Resolução CEPE 2880 de 05/2006: É assegurado a todo aluno regularmente matriculado com frequência mínima de setenta e cinco por cento e média inferior a seis, o direito de ser avaliado por Exame Especial.

Cronograma:

Semana	Conteúdo a ser abordado
20/09/2021 a 24/09/2021 (04 síncronas)	Apresentação da disciplina, trabalho prático, divisão das notas e processos de software
27/09/2021 a 01/10/2021 (04 síncronas)	Gestão de Requisitos, Noções de Git, listas de exercícios e prática do TP- Parte 1
04/10/2021 a 08/10/2021 (04 síncronas)	Gestão de Requisitos, Noções de Git, listas de exercícios e prática do TP- Parte 2

13/10/2021 a 15/10/2021 (04 síncronas)	Fundamentos de teste, exercícios, demandas do TP- Parte 1
18/10/2021 a 22/10/2021 (04 síncronas)	Fundamentos de teste, exercícios, demandas do TP - Parte 2
25/10/2021 a 28/10/2021 (04 síncronas)	Prova (1) (26/10/2021) Projeto e documentação de teste, Verificação Validação; Parte 1
03/11/2021 a 05/11/2021 (02 síncronas)	Projeto e documentação de teste, Verificação Validação; Parte 2
08/11/2021 a 12/11/2021 (04 síncronas)	Fundamentos da Gerência de Configuração de Software
16/11/2021 a 19/11/2021 (04 síncronas)	Vista de prova (1); Medição de Software, Métricas, Relação de Code Smells. Idiomas de Programação - Parte 1
22/11/2021 a 26/11/2021 (04 síncronas)	Medição de Software, Métricas, Relação de Code Smells. Idiomas de Programação. Listas de exercícios - Parte 2
22/11/2021 a 26/11/2021 (04 síncronas)	Qualidade: MPSBR, CMMI & Plano de Qualidade- Parte 1
29/11/2021 a 03/12/2021 (04 síncronas)	Qualidade: MPSBR, CMMI & Plano de Qualidade. Listas de exercícios - Parte 2
06/12/2021 a 10/12/2021 (02 síncronas e 02 assíncronas)	Apresentação do trabalho prático
13/12/2021 a 17/12/2021 (04 síncronas)	Prova (2) (14/12/2021) e vista de prova (2)
11/01/2022 (02 síncronas)	Exame Especial

Bibliografia básica:

- Sommerville, I. Engenharia de Software, 9^a ed., Pearson, 2011.
<https://plataforma.bvirtual.com.br/Acervo/Publicacao/2613>
- VASQUEZ, C. SIMÕES, C. Engenharia de Requisitos, Brasport, 2016.
<https://plataforma.bvirtual.com.br/Acervo/Publicacao/160193>
- Refactoring. Improving the Design of Existing Code. by Martin Fowler, with KentBeck. 2018 <https://refactoring.com/catalog/>
- Sommerville, I. Engenharia de Software, 10^a ed., Pearson, 2019.
<https://plataforma.bvirtual.com.br/Acervo/Publicacao/168127>

GAMMA, Erich. Design patterns elements of reusable object-oriented software / Erich Gamma... [et al.].. Reading: Addison Wesley 1995. 395 p. (Addison-WesleyProfessional Computing Series). ISBN 0201633612 (enc.).

<https://doku.pub/download/padroes-de-projeto-soluoes-reutilizaveis-de-software-orientado-a-objetos-erich-gamma.pdf-6lkvge8em804>

Bibliografia complementar:

- Teste de Software / organizador Henrique Cacique Braga. São Paulo. Pearson Education do Brasil. Ed. 1°. 2016. 139 p. ISBN 9788543020211.
<https://plataforma.bvirtual.com.br/Acervo/Publicacao/150962>
- Qualidade de software. Organizador Giocondo Marino Antonio Gal... Páginas: 139. Editora: Editora Pearson Edição: 1ª. Idioma: Português
<https://plataforma.bvirtual.com.br/Acervo/Publicacao/124148>
- FOWLER, Martin. UML distilled: a brief guide to the standard object modeling language. 3rd ed. Boston: Addison Wesley, c2004. xxx, 175 p. (Object technologyseries). ISBN 0321193687 (Broch.).
<https://integrada.minhabiblioteca.com.br/#/books/9788560031382/cfi/6/2!/4/18/52@0:76.8>