



Plano de Ensino

1 Código e nome da disciplina

ARA0097 ENGENHARIA DE SOFTWARE

2 Carga horária semestral

80

3 Carga horária semanal

4h

4 Perfil docente

O docente desta disciplina deverá ser graduado preferencialmente em Sistemas de Informação, Ciência da Computação, Informática, Engenharias ou áreas afins, e possuir Pós-Graduação Lato Sensu (especialização) preferencialmente na área de Informática. É desejável que o docente possua Pós-Graduação Stricto Sensu (mestrado e/ou doutorado) na área de computação.

É desejável que o docente possua experiência de três anos em docência de nível superior na disciplina, além de conhecimentos teóricos e práticos, habilidades de comunicação em ambiente acadêmico, capacidade de interação e fluência digital para utilizar ferramentas necessárias ao desenvolvimento do processo de ensino-aprendizagem (SGC, SAVA, BdQ e SIA). Importante, também, o conhecimento do Projeto Pedagógico dos Cursos que a disciplina faz parte na Matriz Curricular.

É necessário que o docente domine as metodologias ativas inerentes à educação por competências e ferramentas digitais que tornam a sala de aula mais interativa. A articulação entre teoria e prática deve ser o eixo direcionador das estratégias em sala de aula. Além disto, é imprescindível que o docente estimule o autoconhecimento e autoaprendizagem entre seus alunos.

5 Ementa

FUNDAMENTOS DE SOFTWARE E GERENCIAMENTO DE PROJETOS. FASES DO DESENVOLVIMENTO DE SOFTWARE. MODELOS DE PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE. QUALIDADE DE SOFTWARE. GERENCIAMENTO DE CONFIGURAÇÕES.

6 Objetivos

- Interpretar os conceitos iniciais da engenharia de software, com base na necessidade da sua utilização, para melhor compreender o funcionamento de um software.
- Analisar as etapas para desenvolver um software, com base nas etapas definidas pela engenharia

de software, para entregar um bom produto.

- Distinguir os processos de desenvolvimento de software, com base nos seus métodos e procedimentos, para aprender a modelar e estruturar o desenvolvimento de sistemas.
- Examinar o processo de criação de um sistema, com base nas boas práticas de engenharia de software, para garantir a qualidade do produto.
- Realizar o gerenciamento de configurações, com base nas principais técnicas utilizadas no mercado de trabalho, para manter o controle de versões de software.

7 Procedimentos de ensino-aprendizagem

A disciplina adotará o modelo de aprendizagem baseada em problemas.

O processo de ensino-aprendizagem iniciará por meio de uma situação- problema (problematização/pergunta geradora), previamente definida pela/pelo docente a partir dos temas de aprendizagem. Poderão ser utilizados como estratégias didáticas: exposição, discussão de filmes e documentários, estudos de casos que subsidiarão a análise de problemas, debates estruturados, fóruns de discussão, brainstormings, jogos e ferramentas digitais que tornarão o aluno protagonista de seu aprendizado. Esta abordagem prioriza o aluno, sendo este capaz de articular os temas discutidos nas aulas para responder à situação problema que abre a preleção.

É importante destacar o uso da Sala de Aula Virtual de Aprendizagem (SAVA), local em que o aluno terá acesso ao conteúdo digital da disciplina, poderá resolver questões propostas e explorar conteúdos complementares disponíveis para estudo.

O modelo de aprendizagem prevê ainda a realização da Atividade Prática Supervisionada, que são atividades práticas realizadas em laboratórios, bibliotecas e trabalhos individuais e/ou em grupo que fazem parte do ecossistema de aprendizagem global e local.

8 Temas de aprendizagem

1. FUNDAMENTOS DE SOFTWARE E GERENCIAMENTO DE PROJETOS
 - 1.1 PROCESSO DE FABRICAÇÃO DO HARDWARE X PROCESSO DE DESENVOLVIMENTO DO SOFTWARE
 - 1.2 ENGENHARIA DE SOFTWARE
 - 1.3 O DESENVOLVIMENTO DE SOFTWARE COMO UM PROJETO E AS ATIVIDADES DE GERENCIAMENTO
 - 1.4 PLANEJAMENTO E GERENCIAMENTO DE RISCOS
2. FASES DO DESENVOLVIMENTO DE SOFTWARE
 - 2.1 ENGENHARIA DE REQUISITOS E ANÁLISE DE SISTEMA
 - 2.2 PROJETO DO SISTEMA
 - 2.3 IMPLEMENTAÇÃO E TESTES
 - 2.4 MANUTENÇÃO
3. MODELOS DE PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE
 - 3.1 MODELOS DE PROCESSOS PRESCRITIVOS
 - 3.2 O PROCESSO UNIFICADO
 - 3.3 DESENVOLVIMENTO ÁGIL - EXTREME PROGRAMMING (XP)
 - 3.4 DESENVOLVIMENTO ÁGIL - SCRUM E PROCESSO UNIFICADO ÁGIL

- 4. QUALIDADE DE SOFTWARE (ATIVIDADE PRÁTICA SUPERVISIONADA)
 - 4.1 QUALIDADE DE PROCESSO E DE PRODUTO DE SOFTWARE
 - 4.2 PROCESSO DA GARANTIA DE QUALIDADE DE SOFTWARE
 - 4.3 PLANEJAMENTO DA QUALIDADE E O CONTROLE DA QUALIDADE DE SOFTWARE
 - 4.4 MEDIÇÕES E MÉTRICAS DO SOFTWARE
- 5. GERENCIAMENTO DE CONFIGURAÇÕES
 - 5.1 FUNDAMENTOS DE GERENCIAMENTO DE CONFIGURAÇÕES
 - 5.2 GERENCIAMENTO DE MUDANÇAS, VERSÕES E RELEASES
 - 5.3 CONSTRUÇÃO DE SISTEMAS E FERRAMENTAS CASE

9 Procedimentos de avaliação

Os procedimentos de avaliação contemplarão as competências desenvolvidas durante a disciplina por meio de provas presenciais, denominadas AV e AVS, sendo a cada uma delas atribuído o grau de 0,0 (zero) a 10 (dez) no formato PNI - Prova Nacional Integrada.

Caso o aluno não atinja o resultado desejado na prova de AV, ele poderá recuperar sua nota na prova de AVS. Será composta por uma prova no formato PNI - Prova Nacional Integrada, com total de 10 pontos, e substituirá a nota da AV, caso seja maior.

Para aprovação na disciplina, o aluno deverá, ainda:

- atingir nota igual ou superior a 6 (seis) na prova de AV ou AVS;
- frequentar, no mínimo, 75% das aulas ministradas.

10 Bibliografia básica

ANDRADE, Mayb. **Qualidade de Software**. 1ª Ed. Rio de Janeiro: SESES, 2015.

Disponível em: <https://repositorio2.azurewebsites.net/api/objetos/efetuaDownload/405d3e91-3eef-4471-9ead-702cee3d2861>

PÁDUA FILHO, Wilson de Paula. **Engenharia de software : projetos e processos**. 4a. Rio de Janeiro: LTC - Livros Técnicos e Científicos Editora Ltda., 2019. 2.

Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9788521636748>

PRESSMAN, Roger; MAXIM, Bruce. **Engenharia de Software**. Porto Alegre: AGMH, 2016.

Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9786558040118/>

11 Bibliografia complementar

Amui, Saulo. **Processos de Desenvolvimento de Software**. 1ª Ed.. Rio de Janeiro: SESES, 2015.

Disponível em: <https://repositorio2.azurewebsites.net/api/objetos/efetuaDownload/faf38cab-2fb5-48d6-ac0b-a2685e2f5f48>

KECHI, Hirama. **Engenharia de software: qualidade e produtividade com tecnologia**. 1a. Rio de Janeiro: Elsevier Editora Ltda., 2011.

Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9788595155404/>

MORAIS, Izabelly Soares de; ZANIN, Aline. **Engenharia de Software**. 1ª. Porto Alegre: SAGAH, 2017.

Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9788595022539/>

SCHACH, Stephen R. **Engenharia de Software - Os paradigmas clássicos orientados a objetos**. 7ª Ed. Porto Alegre: ArtMed, 2010.

Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788563308443/>

VETTORAZZO, Adriana de Souza. **Engenharia de Software**. 1ª. São Paulo: SAGAH Educação, 2018.

Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9788595026780/>