

EMENTA

PLANO DE ENSINO

1. IDENTIFICAÇÃO

Campus: Avançado Quedas do Iguaçu

Eixo tecnológico: Informação e Comunicação

Curso: Técnico em Informática Integrado ao Ensino Médio

Componente curricular: Engenharia de Software

Docente: Odair Moreira de Souza

Carga horária: 80 horas/aula e 67 horas/relógio

Turno: Matutino e vespertino

Número de aulas na semana: 2 aulas - 1,67 horas

Período letivo: 2020

Turma (s): 2º A

Coordenador do curso: Odair Moreira de Souza

2. EMENTA

Introdução à Engenharia de Software. Processos de Desenvolvimento de Software. Desenvolvimento Ágil de Software. Engenharia de Requisitos. Modelagem de Sistemas. Introdução à UML. Análise e Projeto de Sistemas. Projeto orientado a objetos com UML. Padrões de projeto. Padrões de arquitetura. Testes de software. Manutenção de Software. Introdução ao Gerenciamento e Planejamento de projetos de software. Gerenciamento de Configuração. Introdução a Qualidade de Software.

3. OBJETIVOS

3.1 Objetivo geral

Compreender os conceitos de engenharia de software, processos de desenvolvimento de software, modelagem de projetos de software, técnicas de teste de software e avaliação da qualidade de produtos e processos.

3.2 Objetivos específicos

- Conhecer os conceitos fundamentais da engenharia de software;
- Compreender as metodologias de desenvolvimento de software;
- Aprender a elaboração de engenharia de requisitos;

- Conhecer e utilizar a linguagem de especificação UML;
- Utilizar uma ferramenta CASE para a análise e projeto;
- Analisar e projetar softwares orientados a objetos;
- Entender e implementar os principais tipos de teste de software;
- Aprender as abordagens para o planejamento e gerenciamento de projetos de software; e,
- Conhecer as medidas de qualidade de software.

4. CONTEÚDO PROGRAMÁTICO

- Introdução à Engenharia de Software
 - Conceitos de Processo e Produto de Software
 - Etapas do Processo de Desenvolvimento de Software
 - Ciclo de Vida de Software
- Processo de Desenvolvimento de Software
 - Modelos de Processo Prescritivo
 - O Processo Unificado
- Desenvolvimento Ágil
 - Método XP
 - Scrum
- Engenharia de Requisitos
 - Requisitos Funcionais e Não-Funcionais
 - Processo de Especificação de Requisitos
 - Especificações de Requisitos de Usuário
 - Documentos de Requisitos
- Prototipagem de Interfaces de Software
- Modelagem de Software
 - Linguagem UML
 - Projetos de Arquitetura
 - Projeto de Projetos
- Teste de Software
 - Verificação e Validação de Software
 - Processo de Teste de Software
 - Tipos de Testes de Software

- Teste de caixa-branca e caixa-preta
- Gerenciamento e Planejamento de Software
 - Gerenciamento de Projetos de Software
 - Planejamento de Projetos de Software
 - Gerenciamento de Configuração
 - Manutenção de Software
- Introdução a Qualidade de Software
 - Garantia da Qualidade de Software
 - Qualidade de Produto e Processo
 - Modelos de Qualidade de Produto
 - Melhorias de Processo de Software

5. METODOLOGIA DE ENSINO E AVALIAÇÃO

A metodologia a ser empregada no processo de ensino-aprendizagem consiste em um conjunto de métodos e estratégias didáticas, tais como, aula expositiva-dialogada, mapas conceituais, tempestade cerebral, soluções de problemas, pesquisa de campo, estudo de caso, estudos com pesquisa, exposições, estudos dirigidos, aulas práticas incentivando a participação dos discentes durante a explanação do conteúdo, resolução de problemas reais, construção e contextualização dos conhecimentos abordados.

O conteúdo teórico será abordado utilizando como referencial teórico a bibliografia sugerida, por meio de exemplos práticos e estudos de caso e ao final de cada assunto, ou conjunto de assuntos relacionados, será proposto aos discentes o desenvolvimento de exercícios individuais ou em equipes para fixação da teoria apresentada, os alunos irão realizar apresentações das soluções desenvolvidas em modelo de arguição, somente para os professores, ou exposição para a turma.

As aulas serão práticas, ministradas no laboratório de informática e para todos os conteúdos abordados, serão apresentados exemplos desenvolvidos em ambiente computacional desenvolvendo exemplos e exercícios. Em seguida solicita-se aos discentes que apliquem os conceitos expostos, com o intuito de incentivar a reflexão e a habilidade de raciocínio para resolução de problemas.

O processo de avaliação para verificar se os objetivos estão sendo atingidos será contínuo e contemplará todas as atividades desenvolvidas, tais como, aplicação de exercícios, avaliações (provas), atividades extras, elaboração de projetos de soluções computacionais, avaliação de atividades em grupos e individuais, atividades de pesquisas e observações periódicas para avaliação longitudinal.

Nesse componente curricular os discentes irão desenvolver um projeto de software (com escolha livre e orientada pelo docente), desde o levantamento de requisitos, análise, modelagem do software, projeto arquitetural, gerenciamento das atividades, projeto de teste, análise de qualidade e a aplicação de um processo de desenvolvimento de software ágil; nesse projeto os discentes terão vários papéis, tais como, analista de requisitos, engenheiro de software, analista de teste, gerente de projetos e entre outros. Além de ministrar as aulas o docente atuará nos projetos como cliente levantando as demandas de requisitos, validação de artefatos e gerente de projetos, orientando o projeto real de engenharia de software das equipes.

5.1 Recursos didáticos

Será utilizado laboratório de informática, projetor multimídia e plataformas online de ensino-aprendizagem de programação.

6. CRITÉRIOS DE AVALIAÇÃO

A avaliação será realizada seguindo as orientações da Resolução nº 50/2017, que estabelece as normas de avaliação dos processos de ensino e aprendizagem no IFPR. Em cada bimestre, o conceito do aluno será composto por meio de sua participação e desempenho nos seguintes pontos:

- Participação e desempenho no desenvolvimento dos trabalhos individuais e em equipes;
- Desenvolvimento de aplicações práticas;
- Desempenhos em avaliações individuais teóricas e práticas;
- Participação, assiduidade e proatividade.

Considera-se que para cada bimestre serão aplicados, no mínimo quatro atividades avaliativas prática de desenvolvimentos (40% do conceito bimestral) e uma ou duas avaliações teóricas/práticas (60% do conceito bimestral). Ressalta-se que os trabalhos e listas devem ser apresentados para a turma e/ou somente para o professor em formato de arguição, se for para a turma será aberto para debate sobre a solução algorítmica desenvolvida.

Os resultados serão apresentados a cada atividade avaliativa, sendo explicitado o diagnóstico feito pelo docente. De acordo com as normas da Instituição, os alunos receberão os conceitos A, B, C ou D nos períodos determinados pelo IFPR e no final do conteúdo de cada área curricular.

Interdisciplinaridade

Essa disciplina poderá ter atividades avaliativas interdisciplinares, envolvendo conteúdos relacionados com as disciplinas de Física, Fundamentos de Informática e Matemática por meio de atividade avaliativa prática de desenvolvimento.

Crítérios de Aprovação

Conforme previsto na Resolução nº 50/2017 do IFPR, Art. 16. A aprovação dos estudantes ocorrerá considerando os seguintes critérios:

I – obtenção de conceito A, B ou C no componente curricular e frequência igual ou superior a 75% (setenta e cinco por cento) da carga horária total no período letivo dos cursos técnicos de nível médio.

7. RECUPERAÇÃO

I - Programa de Atividades e de Orientação: A recuperação do estudante poderá ser realizada no decorrer do módulo da disciplina através de atendimento mais direto e individualizado e com atividades diferenciadas que possam contemplar sua aprendizagem, baseada na Resolução nº 50/2017 do IFPR. O aluno com conceito insuficiente em um bimestre será convocado a participar de recuperação paralela contínua durante o próximo bimestre, exceto no último bimestre, pois isso possibilitará ao alunos tempo e atendimento para compreender a conteúdo antes da avaliação de recuperação. Caso a aprendizagem ainda for considerada insuficiente, o estudante cursa a disciplina novamente como progressão, em horários previamente combinados.

II - Formas de Avaliação: As atividades avaliativas de recuperação serão ofertadas em contraturno dos estudantes e em dias que não tem aula. Além disso, os alunos contam com atendimentos individualizados pelo professor. As recuperações paralelas serão agendadas com no mínimo 15 (quinze) dias de antecedência da aplicação.

III - Direito de Realizar as Avaliações de Recuperação: Conforme Resolução CONSUP/IFPR nº 50/2017, serão oferecidos estudos de recuperação paralela ou retomada dos conteúdos a todos os estudantes, independente do conceito atingido ser B, C ou D.

8. BIBLIOGRAFIA

8.1 Bibliografia Básica

1. FILHO, W. P. P. Engenharia de Software: Fundamentos, Métodos e Padrões. 3.d. Rio de Janeiro: LTC, 2015.
2. PRESSMAN, R. S.; MAXIM, B. R. Engenharia de Software: Uma Abordagem Profissional. 8. ed. Porto Alegre: AMGH, 2016.
3. SOMMERVILLE, I. Engenharia de Software. 9. ed. São Paulo: Pearson Prentice Hall, 2011.
4. ENGHOLM JR, H. Engenharia de Software na prática. São Paulo: Novatec, 2010.
5. BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML: Guia do Usuário. 2. ed. Rio de Janeiro: Elsevier, 2012.

8.2 Bibliografia Complementar

1. BEZERRA, E. Princípios de análise e projetos de sistemas com UML. 3. ed. Rio de Janeiro: Elsevier, 2015.
2. CARDOSO, C. UML na prática: do problema ao sistema. Rio de Janeiro: Ciência Moderna, 2003.
3. DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. Introdução ao teste de software. Rio de Janeiro, RJ: Elsevier, 2007. 394 p. ISBN 9788535226348.
4. COHN, M. Desenvolvimento de software com scrum: aplicando métodos ágeis com sucesso. Porto Alegre: Bookman, 2011. 496 p. ISBN 9788577808076.
5. PRIKLADNICKI, R.; WILLI, R.; MILANI, F.; Métodos Ágeis Para Desenvolvimento De Software. Editora: Bookman, 2014. ISBN: 9788582602072

9. OBSERVAÇÕES

- A distribuição do conteúdo das aulas é uma previsão e poderá ser adequado durante o andamento do ano letivo para atender aos reajustes em função do desempenho da turma ou para tratar eventualidades.
- O Google Classroom será o canal de comunicação digital entre o docente e os discentes, para disponibilização dos materiais das aulas, listas de exercícios, implementações exemplos, submissão de atividades, avisos para a turma, agendamento de atividades.
- O acesso dos estudantes aos laboratórios para o desenvolvimento de atividades em horários alternativos aos das aulas e atendimentos deverá ser solicitado ao professor e ter anuência do responsável pelo laboratório.
- O local e os horários de atendimentos do docente e do monitor de programação serão divulgados nos murais de avisos do campus e na seção de avisos para a turma no Google Classroom.



Documento assinado eletronicamente por **ODAIR MOREIRA DE SOUZA, Servidor Docente**, em 08/03/2020, às 09:58, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://sei.ifpr.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0696002** e o código CRC **F3845DED**.

INSTITUTO FEDERAL DO PARANÁ | QUEDAS/SENS/QUEDAS/DG/IFPR/QUEDAS/IFPR/CASCADEL-SENS/QUEDAS
Rua Emilio Bertolini, nº 54, Curitiba - PR | CEP CEP 82920-030 - Brasil