



Plano de Disciplina

Identificação				
CURSO Mestrado Profissional em Tecnologia da Informação			CAMPUS João Pessoa	
DISCIPLINA Teste de Software			CÓDIGO DA DISCIPLINA 30632	
PRÉ-REQUISITO Não há pré-requisito				
UNIDADE CURRICULAR (OBRIGATÓRIA, OPTATIVA, ELETIVA) Eletiva			SEMESTRE 2019.2	
DOCENTES RESPONSÁVEIS Edemberg Rocha Crishane Freire				
Carga Horária				
TEÓRICA 24	PRÁTICA 24	EaD ¹ 12	CARGA HORÁRIA SEMANAL: 3	CARGA HORÁRIA TOTAL 60

Ementa

Princípios e técnicas de testes de software: teste de unidade; teste de integração; teste de regressão; teste funcional. Desenvolvimento orientado a testes. Automação dos testes. Geração de casos de teste. Testes alfa, beta e de aceitação. Ferramentas de testes. Planos de testes. Gerenciamento do processo de testes. Registro e acompanhamento de resultados de testes.

Objetivos

Conhecer os conceitos relacionados a teste de software em diversos ambientes e situações; identificar desafios para pesquisa e inovação em teste de software por meio da discussão de problemas em aberto e de demandas por soluções para problemas práticos advindos da indústria e da academia.

Conteúdo Programático - Presencial

Tópico	Carga horária
Introdução a Teste de Software - Visão Geral e História	3
Princípios e técnicas de testes de software	3
Análise e Discussão de Pesquisa em Teste de Software: Artigos	21
Desafios e Tendências em Teste de Software: Seminários	6
Aplicação em Teste de Software (métodos, técnicas, ferramentas e processos)	15

Conteúdo Programático - EAD

Tópico	Carga horária
Orientação para o Projeto e Escrita de Artigo	6
Orientação para Preparação do Seminário	6

¹ Para a oferta de disciplinas na modalidade à distância, integral ou parcial, desde que não ultrapasse 20% (vinte por cento) da carga horária total do curso, observar o cumprimento da Portaria MEC nº 1.134, de 10 de outubro de 2016.



Metodologia de Ensino

Presencial

Aulas teóricas e práticas.

EAD

Planejamento de projeto de pesquisa utilizando a ferramenta de colaboração institucional. Entregas de etapas da pesquisa seguindo deadlines pré-definidos utilizando envios de arquivos na plataforma.

Planejamento de seminário utilizando a ferramenta de colaboração institucional. Entregas de etapas do seminário seguindo deadlines pré-definidos utilizando envios de arquivos na plataforma.

Recursos Didáticos

Quadro, projetor, ferramentas de software open source, periódicos, artigos, acesso à Internet

Avaliação

A avaliação será baseada nas seguintes atividades:

- Projeto e artigo (50%);
- Seminário (30%) – avaliado pelos professores e alunos;
- Exercícios e participação (20%).

Bibliografia

1. DELAMARO, M.E.; MALDONADO, J. C.; JINO, M. Introdução ao Teste de Software. Rio de Janeiro: Campus, 2007.
2. MOLINARI, L. Inovação e Automação de Testes. São Paulo: Erica, 2010. BECK, Kent. TDD Desenvolvimento Guiado por Testes. Porto Alegre: Bookman, 2010.
3. PEZZE, M.; YOUNG, M. Software Testing and Analysis: Process, Principles and Techniques. [S.l.]: Wiley, 2007.
4. PEZZE, M.; YOUNG, M. Teste e Análise de Software. Porto Alegre: Bookman, 2008.
5. LINZ, T. Testing in Scrum: A Guide for Software Quality Assurance in the Agile World. 1. ed. [S.l.]: Rocky Nook Computing, 2014.
6. BLACK, R. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing. [S.l.]: Wiley, 2009.
7. WHITAKKER, J.A.; ARBON, J.; CAROLLO, J. How Google Tests Software. [S.l.]: AddisonWesley Professional, 2012.
8. OBERKAMPF, W.L.; ROY, C.J. (2010). Verification and Validation in Scientific Computing. Cambridge University Press. pp. 154–5. ISBN 9781139491761.

Observações

(Nenhuma)