#### **Teste de Software**

Carga horária

Teórica: 32 horas

Prática: 32 horas

Total: 64 horas

Unidade responsável: Instituto de Informática (INF)

Natureza: Núcleo Específico (NE) / Obrigatória

Pré-requisito: Não possui

Ementa:

1. Processo de construção (8h): definições básicas, atividades e documentação.
2. Processo de Teste de Software (56): definições básicas, técnicas de teste, teste baseado em intuição e experiência do engenheiro de software, atividades do processo, documentação e ferramentas.

#### *Condições mínimas (estar apto a)*

* (Faz sem orientação). Explicar o teste de software, a relação dessa área de conhecimento com as demais da engenharia de software e a relação com o conceito de sistema.
* (Segue instruções). Usar um processo de teste de software.
* (Segue instruções). Desenvolver planos de teste para o teste de unidade.
* (Faz com orientação). Propor casos de teste segundo algum critério da técnica de teste funcional para o teste de unidade.
* (Faz com orientação). Propor casos de teste segundo algum critério da técnica de teste estrutural para o teste de unidade.
* (Faz com orientação). Construir código para automação do teste de software.
* (Segue instruções). Aplicar teste exploratório no teste de sistema.
* (Faz sem orientação). Executar casos de teste.
* (Faz sem orientação). Citar e explicar métricas de teste de software.
* (Segue instruções). Realizar medições pertinentes a teste de software.
* (Faz com orientação). Documentar atividades do teste de software.
* (Segue instruções). Identificar e utilizar ferramentas de teste de software.

Bibliografia básica:

* MYERS, G. J. The Art of Software Testing. Wiley. 2011.
* COPELAND, L. A Practitioner’s Guide to Software Test Design. Artech House, 2003.
* SYKES, D. A.; MCGREGOR, J. D. A practical guide to testing object-oriented software. Addison-Wesley, 2001. ISBN 0201325640.

Bibliografia complementar:

* MESZAROS, G. xUnit test patternsrefactoring test code. Addison-Wesley, 2007. ISBN 0131495054.
* MOSLEY, D. J. Client-server software testing on the desktop and the Web. Prentice Hall, 2000. ISBN 0131838806.
* GROSS, H. Component-based software testing with UML. Springer, 2005. ISBN 354020864X.
* PERRY, W. E. Effective methods for software testing. 2nd edition. J. Wiley, 1999. ISBN 047135418X.
* WU, M.; LI, K. Effective software test automationdeveloping an automated software testing tool. SYBEX, 2004. ISBN 0782143202.

Informações: