

# Detalhes do Plano de Ensino

|                    |  |
|--------------------|--|
| Código             | CMP1492  |
| Nome da Disciplina | Gerência de Qualidade de Software  |
| Carga Horária      | 60   |
| Créditos           | 4  |
| Ementa             | Qualidade de software: verificação, validação e testes. Tipos de revisões: revisão técnica formal, inspeção, walkthroughs, auditorias. Estratégias de V&V. Tipos de testes, estratégias de testes e processo de garantia de qualidade de software do MPS-BR. |

## Objetivos Gerais

- Estabelecer fundamentação teórica sobre verificação Diferenciar validação de testes
- Conhecer os tipos de revisões formais
- Conhecer as estratégias de V&V
- Diferenciar e entender os tipos de testes
- Capacitar os alunos a montar estratégias de testes

## Objetivos Específicos

- Apresentar a visão geral de qualidade de software
- Destacar a importância de processos de qualidade
- Apresentar aos alunos os conceitos fundamentais de Verificação, Validação e Testes Familiarizar o aluno com as diversas formas de se fazer as atividade de V&V Diferenciar erro de defeito e falha
- Destacar aos alunos as principais técnicas de testes de software
- Apresentar aos alunos as principais tendências em termos V&V e testes.

## Conteúdo Programático

- MPB.BR
- Verificação e Validação
- Revisões Formais
- Técnicas de Revisão Estática
- Introdução ao Teste de Software
- Testes de Unidade, Integração e de Sistema Teste Funcional
- Teste Estrutural C
- ritérios de Teste
- Tipos de Testes
- Teste de Mutação

## Atividades Externas da Disciplina (AED)

**Objetivo da Atividade:** Estimular que os alunos utilizem a pesquisa para automatizar tarefas referente a qualidade de software.

**Descrição da Atividade:** Os alunos devem pesquisar ferramentas que automatizam alguma parte do processo de qualidade de software, avaliar a ferramenta e emitir um relatório de recomendação e utilização da ferramenta.

**Cronograma:** 27/11 - 2 horas serão usadas para orientação dos alunos e explicação da especificação do trabalho de AED. A AED deverá realizada no laboratório ou em casa em horários diferentes das aulas. As orientações poderão ser realizadas nas aulas durante o semestre ou por email diretamente ao professor. O prazo final para a entrega e apresentação da atividade ao professor será dia 27/11.

**Forma de Registro:** Os alunos deverão entregar ao professor um relatório técnico apresentando sua conclusão a respeito da pesquisa realizada, buscando expor os pontos positivos e negativos do processo automatizado relacionando com o conteúdo apresentado em sala de aula..

**Crítérios de Avaliação:** Organização e clareza na escrita da conclusão a respeito do assunto abordado no trabalho

#### Bibliografia de Consulta

- SWEBOK, Guide to the software Engineerig body of Konwledge. IEEE Computer society, 2004 version.
- Wiegers, K. Peer Reviews in Software. Addison-Wesley, 2002.
- Gilb, Tom and Grahnan, Dorothy. Software Inspection. Addison-Wesley, 2002.
- Pressman, Roger S. - "Software Engineering – A Practioner's Approach", McGraw-Hill.
- Sommerville, Ian. Engenharia de Software. Addison Wesley, 2003, 6a Edição

#### Metodologia

- Aprendizado Baseado em Problemas (PBL)
- Aulas expositivas em ambiente virtual de aprendizagem.
- Trabalho em grupos.
- Estudo dirigido.
- Resolução de exercícios em classe.

#### Avaliação

A nota final, NF, da disciplina será resultante da média ponderada de dois conjuntos de notas, N1 e N2, conforme a expressão  $NF = 0,4.N1 + 0,6. N2$ , sendo que tanto N1 quanto N2 serão compostos por no mínimo duas notas resultantes de duas avaliações individuais com todo o conteúdo do período correspondente. Serão aplicados pequenos testes em sala ou trabalhos, cuja soma de suas notas irão compor N1 e N2.

A frequência será computada em cada encontro ou através de chamada feita durante as aulas.

Será considerado aprovado na disciplina o aluno que obtiver a frequência mínima de 75% e Nota Final igual ou superior a seis pontos.

A N1 será calculada com a seguinte formula: 90% da Média dos PBLs do período + 10% da AED parcial.

A N2 será calculada com a seguinte formula: 90% da Média dos PBLs do período + 10% da AED final.

A N Final será calculada com a seguinte formula:  $N1*0,4 + N2*0,6$

Onde, PBL: Problemas para resolução

#### Cronograma

### Dias de Aulas (2 horas-aula cada encontro)

18/08/20 Apresentação e discussão do Plano de Ensino  
21/08/20 Coneitos de Qualidade  
25/08/20 Coneitos de Qualidade  
28/08/20 MPS.BR - Modelos de Referência  
01/09/20 PBL de MPS.BR  
04/09/20 PBL de MPS.BR  
08/09/20 Revisão  
11/09/20 PBL de Coneitos de Qualidade. MPS.BR  
15/09/20 PBL de Verificação e Validação  
18/09/20 PBL de Verificação e Validação  
22/09/20 PBL de Revisões Formais  
25/09/20 PBL de Revisões Formais  
29/09/20 PBL de Técnicas de revisão estática  
02/10/20 PBL de Técnicas de revisão estática  
06/10/20 PBL de Introdução ao teste de software  
09/10/20 PBL de Introdução ao teste de software  
13/10/20 Revisão  
16/10/20 PBL de Testes de Unidade, Integração e Sistema  
20/10/20 PBL de Testes de Unidade, Integração e Sistema  
23/10/20 PBL de Teste Funcional  
27/10/20 PBL de Teste Funcional  
30/10/20 PBL de Técnicas de Teste Funcional  
03/11/20 Revisão  
06/11/20 PBL de Técnicas de Teste Funcional  
10/11/20 PBL de Teste Estrutural  
13/11/20 PBL de Teste Estrutural  
17/11/20 PBL de Critérios baseados em complexidade  
20/11/20 PBL de Critérios baseados em complexidade  
24/11/20 PBL de Critérios baseados em fluxo de controle  
27/11/20 PBL de Critérios baseados em fluxo de controle e AED  
01/12/20 PBL de Critérios baseados em fluxo de dados  
04/12/20 PBL de Critérios baseados em fluxo de dados  
08/12/20 PBL de Desenvolvimento de Plano de Gerenciamento da Qualidade  
11/12/20 PBL de Desenvolvimento de Plano de Gerenciamento da Qualidade  
15/12/20 Discussão de Notas N2

### Bibliografia Básica

1. DELAMARO, M.; MALDONADO, J. C.; JINO, M. Introdução ao teste de software. Rio de Janeiro: Elsevier, 2007.
2. FREEDMAN, D. P.; WEINBERG, G..M. Manual de walkthroughs: inspeções e revisões técnicas de especificações de sistemas e programas. São Paulo: Makron Books do Brasil, c1993.
3. MYERS, G. J. et al. The art of software testing. 3rd ed. Hoboken: J. Wiley & Sons, c2012

### Bibliografia Complementar

1. CHUNG, L. et al. Non-functional requirements in software engineering. Boston: Kluwer Academic, c2000.
2. OULD, Martyn A.; UNWIN, C. (Ed). Testing in software development. Cambridge: Cambridge University Press, c1986
3. PRESSMAN, R. S. Engenharia de software. 6. ed. São Paulo: McGraw-Hill, 2006.
4. SCHWABER, K; BEEDLE, M. Agile software development with scrum. Upper Saddle River: Prentice Hall, 2002.
5. SOMMERVILLE, I. Engenharia de software. 8. ed. São Paulo: Pearson Education do Brasil, 2007.

### Material de Apoio

1. Burnstein, Ilene; Practical software testing. ISBN: 0-387-951
2. Myers, I. Fundamental Software Test.
3. Delamaro, Maldonado e Jino; Introdução ao Teste de Software, Ed. CAMPUS.