

Unidade Curricular	Testes e Qualidade de Software	Área Científica	Ciências Informáticas
CTeSP em	Desenvolvimento de Software	Escola	Escola Superior de Tecnologia e de Gestão de Bragança
Ano Letivo	2022/2023	Ano Curricular	2
Nível	0-2	Créditos ECTS	3.0
Tipo	Semestral	Semestre	1
Código	4088-713-2108-00-22		
Horas totais de trabalho	81	Horas de Contacto	T - TP 10 PL 20 TC - S - E - OT - O -

T - Ensino Teórico; TP - Teórico Prático; PL - Prático e Laboratorial; TC - Trabalho de Campo; S - Seminário; E - Estágio; OT - Orientação Tutoria; O - Outra

Nome(s) do(s) docente(s) Evandro Pires Alves, Maria João Tinoco Varanda Pereira

### Resultados da aprendizagem e competências

No fim da unidade curricular o aluno deve ser capaz de:

1. conhecer as principais normas de qualidade de software
2. conhecer os principais tipos e técnicas de testes de software
3. saber desenvolver testes para um produto de software gerindo o tipo de testes adequado a cada fase do projeto
4. conhecer as principais técnicas de análise estática e de integração de testes unitários do desenvolvimento de aplicações móveis

### Pré-requisitos

Antes da unidade curricular o aluno deve ser capaz de:

1. utilizar conhecimentos básicos de matemática, formalismos e notações matemáticas
2. compreender os conceitos da programação orientada a objetos
3. ser capaz de implementar páginas HTML com recurso a CSS e JavaScript

### Conteúdo da unidade curricular

Normas e Padrões de Qualidade de Software; Fundamentos de Testes; Técnicas de conceção de testes; Gestão de testes; Análise estática e revisões de código; Testes Unitários

### Conteúdo da unidade curricular (versão detalhada)

1. Normas e Padrões de Qualidade de Software - ISO/IEC 25000 - SQuARE
  - requisitos e avaliação de qualidade de sistemas e software
  - critérios de classificação de qualidade do produto de software
2. Fundamentos de Testes
  - níveis e tipos de testes: caixa branca vs caixa preta
  - estruturais (fluxo de dados, métricas de código)
  - funcionais (unitários, de integração, de interface e aceitação)
  - não funcionais (de carga e performance e de segurança)
3. Técnicas de conceção de testes
4. Gestão de testes
  - planeamento de testes durante o ciclo de vida de desenvolvimento do produto
  - ferramentas de gestão de testes e de suporte aos testes
5. Análise estática e revisões de código
6. Testes Unitários e Funcionais
  - PHPUnit para PHP
  - Codeception

### Bibliografia recomendada

1. Farrell-Vinay, Manage Software Testing: Taylor & Francis, 2008.
2. S. Naik and P. Tripathy, Software Testing and Quality Assurance: Theory and Practice, Wiley, 2008.
3. Paul Ammann and Jeff Offutt, Introduction to Software Testing, Cambridge University Press, 2016.
4. Stephen H. Kan, Metrics and Models in Software Quality Engineering, Addison-Wesley Professional, 2003.
5. R. Pressman and B. Maxim, Software Engineering: A Practitioner's Approach, 8th ed. McGraw-Hill Science/Engineering/Math, 2014.

### Métodos de ensino e de aprendizagem

Serão usados o método expositivo e interrogativo (para transmissão de conhecimentos teóricos), a resolução prática de exercícios e o estudo individual, de grupo e acompanhado. Os exercícios a resolver, na aula e fora dela, serão representativos de casos reais simplificados. O material será disponibilizado através do serviço de ensino à distância do IPB.

### Alternativas de avaliação

1. Opção 1 - (Ordinário, Trabalhador) (Final, Recurso, Especial)
  - Trabalhos Práticos - 40%
  - Exame Final Escrito - 60%
2. Opção 2 - (Ordinário, Trabalhador) (Final, Recurso, Especial)
  - Exame Final Escrito - 100%

### Língua em que é ministrada

Português, com apoio em inglês para alunos estrangeiros

### Validação Eletrónica

Evandro Pires Alves, Maria João Tinoco Varanda Pereira	José Luís Padrão Exposto	Paulo Alexandre Vara Alves	Nuno Adriano Baptista Ribeiro
28-09-2022	30-09-2022	07-11-2022	22-11-2022