

Unidade Curricular	Laboratórios de Engenharia de Software		Área Científica	Ciências da Computação	
Licenciatura em	Informática de Gestão		Escola	Escola Superior de Tecnologia e de Gestão de Bragança	
Ano Letivo	2022/2023	Ano Curricular	3	Nível	1-3
Tipo	Semestral	Semestre	1	Créditos ECTS	6.0
Código	9186-709-3103-00-22				
Horas totais de trabalho	162	Horas de Contacto	T 30	TP 30	PL -
			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) Paulo Jorge Teixeira Matos

Resultados da aprendizagem e competências

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

1. selecionar e aplicar o modelo de desenvolvimento que melhor se adequa a um dado projeto
2. desenhar software fazendo uso de conceitos e princípios de desenho orientado aos componentes potenciando a reutilização e manutenção
3. selecionar e aplicar padrões arquiteturais de software em consonância com as necessidades de cada aplicação
4. fazer uso de padrões de desenho na construção de software
5. conceber, desenvolver e fazer uso de frameworks
6. aplicar conceitos de gestão de projetos no processo de desenvolvimento de software

Pré-requisitos

Antes da unidade curricular o aluno deve ser capaz de:

Programar numa linguagem orientada aos objetos ou funcional, preferencialmente Java ou Javascript.

Conteúdo da unidade curricular

Gestão de projetos de software. Ambientes e ferramentas de suporte ao processo de desenvolvimento de software. Arquiteturas de software. Desenho de software. Padrões de desenho e boas práticas no desenvolvimento de software. Verificação e validação de software. Manutenção de software.

Conteúdo da unidade curricular (versão detalhada)

1. Introdução
 - Processo de desenvolvimento de software
2. Conceitos introdutórios de gestão de projetos de software
 - Gestão de equipas
 - Planeamento e gestão de tarefas
 - Ferramentas para a gestão de projetos
3. Ambientes e ferramentas de suporte ao processo de desenvolvimento de software
 - Ambientes de desenvolvimento de software
 - Ferramentas para análise de requisitos e de modelação
 - Ferramentas para configuração e gestão de software
4. Arquiteturas de software
 - Pipe + filter
 - Orientadas aos objetos
 - Por camadas
 - Blackboard
 - Máquina de estados
 - Cliente-servidor
 - Peer-to-peer
 - Orientadas aos eventos
 - Push e Pull based
 - Distribuídas
5. Desenho de software
 - Princípios e conceitos fundamentais sobre desenho de software
 - Desenho orientado aos objetos
 - Padrões de desenho
 - Desenho por componentes
 - Reutilização de software e frameworks
 - Prototipagem de software
6. Verificação e validação de software
 - Estratégias de validação e verificação
 - Princípios fundamentais de teste de software
 - Técnicas de teste black-box e white-box
 - Ferramentas de teste
 - Manutenção de software

Bibliografia recomendada

1. "Design Pattern – Elements of reusable object-oriented software", Erich Gamma, Richard Helm, Ralph Johnson e John Vlissides; Addison-Wesley, 1994.
2. "UML – Metodologias e Ferramentas CASE – Volume I", Alberto Silva e Carlos Videira; Centro Atlantico. pt, 2005.
3. "Software Construction", MITOPENCOURSEWARE, 2016, <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-005-software-construction-spring-2016/index.htm>
4. "Designing Software Architectures: A Practical Approach", Cervantes, H. and Kazman, R., Addison-Wesley, 1 Edition, 2016.
5. "Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects", 1st Edition, Arnon Axelrod, APress, 2018

Métodos de ensino e de aprendizagem

Aulas de cariz experimental, em que através de metodologias ativas os alunos analisam, codificam, verificam e validam soluções para problemas reais. Utilizam-se também metodologias expositivas no sentido de complementar o conhecimento dos alunos, necessário à resolução dos problemas. Período não-presencial visa o estudo, planeamento e a conclusão dos trabalhos realizados nas aulas.

Alternativas de avaliação

1. Alternativa 1 - (Ordinário, Trabalhador) (Final, Recurso)
 - Projetos - 60%
 - Exame Final Escrito - 40% (Esta componente tem nota mínima de 7.0 (sete) valores em vinte.)
2. Alternativa 2 - (Trabalhador) (Especial)
 - Exame Final Escrito - 100%
3. Alternativa 3 - (Ordinário) (Especial)
 - Projetos - 40%
 - Exame Final Escrito - 60% (Esta componente tem nota mínima de 7.0 (sete) valores em vinte.)

Língua em que é ministrada

Português, com apoio em inglês para alunos estrangeiros

Validação Eletrónica

Paulo Jorge Teixeira Matos	José Luís Padrão Exposto	José Carlos Rufino Amaro	Paulo Alexandre Vara Alves
16-10-2022	24-10-2022	24-10-2022	24-10-2022