

Ficha de Unidade Curricular – (Versão A3ES 2018-2023)

1. Caracterização da Unidade Curricular.

1.1. **Designação da unidade curricular** (1.000 carateres).
Engenharia de Software / Software Engineering

1.2. **Sigla da área científica em que se insere** (100 carateres).
IC (MEIC), INF (MERCAM)

1.3. **Duração**¹ (100 carateres).
Semestral

1.4. **Horas de trabalho**² (100 carateres).
162

1.5. **Horas de contacto**³ (100 carateres).
T – 37.5, TP – 10, PL – 20

1.6. **ECTS** (100 carateres).
6

1.7. **Observações**⁴ (1.000 carateres).
Comum MEIC, MERCAM

1.7. **Remarks** (1.000 carateres).
Common MEIC, MERCAM

2. **Docente responsável e respetiva carga letiva na Unidade Curricular** (preencher o nome completo) (1.000 carateres).
Luís Filipe Graça Morgado, 67.5 horas

3. **Outros docentes e respetivas cargas letivas na unidade curricular** (1.000 carateres).

4. **Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes)**. (1.000 carateres).

Os estudantes que terminam com sucesso esta unidade curricular serão capazes de:

1. Compreender e descrever os princípios base de desenvolvimento de sistemas, com aplicação específica aos processos de desenvolvimento de software, bem como os métodos de modelação, projecto, implementação e verificação que suportam esses processos de desenvolvimento;
2. Identificar e relacionar as principais características de diferentes processos de desenvolvimento de software em termos de domínios de aplicação, potencialidades e limitações;
3. Utilizar processos e métodos de desenvolvimento sistemáticos, disciplinados e quantificáveis na concepção e desenvolvimento de software;
4. Verificar e testar os sistemas desenvolvidos tendo em conta garantir a qualidade dos produtos finais.

4. **Intended learning outcomes (knowledge, skills and competences to be developed by the students)**. (1.000 characters).

Students who successfully complete this course will be able to:

1. Understand and describe the basic principles of systems development, with specific application to software development processes, and the methods of software modeling, design, implementation and verification that support such development processes;
2. Identify and relate the main features of various software development processes in terms of application domains, capabilities and limitations;
3. Use systematic, disciplined and quantifiable processes and methods in the design and development of software;
4. Verify and test the developed systems ensuring the quality of the final products.

5. **Conteúdos programáticos (1.000 carateres).**
- I. Introdução à análise e desenvolvimento de sistemas.
 - II. Processos de desenvolvimento de software.
 - III. Análise e especificação de requisitos.
 - IV. Modelação e arquitectura de software.
 - V. Implementação, teste e qualidade.

5. **Syllabus (1.000 characters).**
- I. Introduction to systems analysis and development.
 - II. Software development processes.
 - III. Requirements analysis and specification.
 - IV. Software modelling and architecture.
 - V. Implementation, test and quality.

6. **Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular (1.000 carateres).**

Esta disciplina visa o estudo de abordagens sistemáticas, disciplinadas e quantificáveis de desenvolvimento de software (objectivos 1 e 2 concretizados em I e II) desenvolvendo nos alunos a capacidade de realizar a análise de problemas concretos e a capacidade de modelação, projecto, implementação e teste dos sistemas respectivos, tendo por base processos de desenvolvimento sistemáticos (objectivos 3 e 4 concretizados em III, IV e V).

6. **Evidence of the syllabus coherence with the curricular unit's intended learning outcomes (1.000 characters).**

This course aims to study systematic, disciplined and quantifiable approaches to software development (objectives 1 and 2 achieved in I and II) developing in the students the ability to perform the analysis of concrete problems and the ability of modeling, design, implementation and testing of the resulting systems, based on systematic development processes (objectives 3 and 4 realized in III, IV and V).

7. **Metodologias de ensino (avaliação incluída) (1.000 carateres).**

É utilizada uma metodologia de ensino teórico-prática, suportada em projectos desenvolvidos ao longo do semestre. Os resultados de aprendizagem 1, 2 e 3 são avaliados individualmente através de teste escrito (50%). O resultado de aprendizagem 3 é igualmente avaliado com base em trabalhos realizados em grupo, em conjunto com uma discussão final desses trabalhos (50%), o mesmo acontecendo com o resultado de aprendizagem 4.

7. **Teaching methodologies (including assessment) (1.000 characters).**

Theoretical and practical teaching is used, supported by projects developed throughout the semester. The learning outcomes 1, 2 and 3 are individually evaluated by written test (50%). The learning outcomes 3 and 4 are also evaluated based on group work and a final discussion (50%).

8. **Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular (3.000 carateres).**

Os temas correspondentes aos resultados de aprendizagem 1, 2, 3 e 4, na sua componente conceptual, são estudados em aulas teóricas específicas, e concretizados em casos práticos e projectos desenvolvidos ao longo do semestre em aulas teórico-práticas e de modo autónomo pelos alunos.

8. **Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes (3.000 characters).**

The themes related to learning outcomes 1, 2, 3 and 4, in their conceptual component, are studied in specific lectures, and concretized by practical problems and projects developed during the semester in practical classes and independently by students.

9. **Bibliografia de consulta/existência obrigatória (1.000 carateres).**

Roger Pressman, Software Engineering: a Practitioner's Approach, McGraw Hill, 2014
Craig Larman, Applying UML and Patterns: An Introduction to Object Oriented Analysis and Design and Iterative Development, Prentice Hall, 2004

¹ Anual, semestral, trimestral, ...

² Número total de horas de trabalho.

³ Discriminadas por tipo de metodologia adotado (T - Ensino teórico; TP - Ensino teórico-prático; PL - Ensino prático e laboratorial; TC - Trabalho de campo; S - Seminário; E - Estágio; OT - Orientação tutorial; O - Outro).

⁴ Assinalar sempre que a unidade curricular seja optativa.