

---

**Ano Letivo** 2016-17

---

**Unidade Curricular** ANÁLISE E MODELAÇÃO DE SISTEMAS

---

**Cursos** ENGENHARIA INFORMÁTICA (1.º ciclo)

---

**Unidade Orgânica** Faculdade de Ciências e Tecnologia

---

**Código da Unidade Curricular** 14781056

---

**Área Científica** CIÊNCIA DE COMPUTADORES

---

**Sigla**

---

**Línguas de Aprendizagem** Português

---

**Modalidade de ensino** Presencial

---

**Docente Responsável** Marielba Silva de Zacarias

---

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Marielba Silva de Zacarias	PL; T	T1; PL1	30T; 30PL

\* Para turmas lecionadas conjuntamente, apenas é contabilizada a carga horária de uma delas.

ANO	PERÍODO DE FUNCIONAMENTO*	HORAS DE CONTACTO	HORAS TOTAIS DE TRABALHO	ECTS
3º	S1	30T; 30PL	168	6

\* A-Anual;S-Semestral;Q-Quadrimestral;T-Trimestral

#### Precedências

Sem precedências

#### Conhecimentos Prévios recomendados

- Programação Orientada a Objectos
- Engenharia de Software

#### Objetivos de aprendizagem (conhecimentos, aptidões e competências)

- **Objectivo A:** Adquirir conceitos fundamentais de modelação conceptual de forma a preparar aos alunos para a modelação conceptual independentemente da linguagem utilizada,
- **Objectivo B:** Adquirir conceitos da modelação conceptual orientada a objectos
- **Objectivo C:** Aprender a utilizar a linguagem de modelação UML em todas as suas vertentes: estrutura, comportamento e arquitetura,
- **Objectivo D:** Distinguir entre diagramas de análise e diagramas de desenho e
- **Objectivo E:** Dar a conhecer aos alunos a existência de outros paradigmas e linguagens de modelação conceptual (SysML, OWL)

#### Conteúdos programáticos

1. Conceitos introdutórios: UoD, linguagem de modelação, modelo, esquema, processo (metodologia), método (abordagem),
2. Modelação de software: motivação e princípios,
3. Análise e especificação de requisitos (revisão)
4. Introdução ao UML: vertentes funcional, estrutura, comportamento e arquitetura,
5. Diagramas de casos de utilização,
6. Diagramas de classes e objetos,
7. Diagramas de interação, Diagramas de estado e atividades,
8. Diagramas de componentes, Diagramas de instalação
9. Outras linguagens (SysML, RDF e OWL).

### **Metodologias de ensino (avaliação incluída)**

#### **Aulas Teóricas (30 horas)**

As noções teóricas serão dadas por método predominantemente expositivo, com projeção e explicação dos objectivos e conteúdos correspondentes a cada tema, acompanhado de debate, colocação e esclarecimento de dúvidas. Os alunos também têm oportunidade de aprofundar os conhecimentos expostos através de um web site desenhado especificamente para a disciplina.

#### **Aulas Práticas (30 horas)**

Os estudantes serão motivados para aplicar as competências adquiridas através de exercícios práticos de pequena dimensão e depois através da execução de um projeto que executam ao longo da disciplina..

#### **Componentes de avaliação**

- Exame: 50%
  - Projecto: 35%
  - Exercícios: 15%
- 

### **Bibliografia principal**

Recursos fornecidos em <http://w3.ualg.pt/~mzacaria/tutorial-uml/index.html>

**Academic Year** 2016-17

**Course unit** SYSTEMS ANALYSIS AND MODELLING

**Courses** INFORMATICS (COMPUTER SCIENCE) (1st Cycle)

**Faculty / School** Faculdade de Ciências e Tecnologia

**Main Scientific Area** CIÊNCIA DE COMPUTADORES

**Acronym**

**Language of instruction** Portuguese

**Learning modality** Presencial

**Coordinating teacher** Marielba Silva de Zacarias

Teaching staff	Type	Classes	Hours (*)
Marielba Silva de Zacarias	PL; T	T1; PL1	30T; 30PL

\* For classes taught jointly, it is only accounted the workload of one.

#### Contact hours

T	TP	PL	TC	S	E	OT	O	Total
30	0	30	0	0	0	0	0	168

T - Theoretical; TP - Theoretical and practical ; PL - Practical and laboratorial; TC - Field Work; S - Seminar; E - Training; OT - Tutorial; O - Other

#### Pre-requisites

no pre-requisites

#### Prior knowledge and skills

- Object Oriented Programming Concepts and Practice
- Software Engineering Concepts and Practice

#### The students intended learning outcomes (knowledge, skills and competences)

- **Outcome A** : Learn basic concepts of conceptual modeling in order to prepare students for conceptual modeling regardless of the language used,
- **Outcome B**: Learn object-oriented conceptual modelling concepts,
- **Outcome C**: Learn to use all perspectives of the UML modeling language: structure, behavior and architecture,
- **Outcome D**: Distinguish between analysis diagrams and drawing diagrams
- **Outcome E**: Acquaint students with the existence of other paradigms and conceptual modeling language (SysML, OWL)

#### Syllabus

1. Introductory concepts: UOD, language modeling, design, layout, process (methodology), method (approach)
2. Software Modeling: motivation, importance and principles,
3. Requirement analysis and specification (review)
4. Introduction to UML: functional aspects, structure, behavior and architecture
5. Use case diagrams
6. Class and Object Diagrams
7. Interaction Diagrams, State and Activity Diagrams
8. Component Diagrams, Deployment diagrams
9. Other modelling languages (SysML, RDF and OWL)

### **Teaching methodologies (including evaluation)**

#### **Lectures (30 hours)**

Theoretical notions are given by oral expositions, with projection and explanation of the objectives and contents corresponding to each theme, followed by debate, and question answering. Students also have the opportunity to deepen their knowledge exposed through a web site designed specifically for the discipline.

#### **Practical sessions (30 hours)**

Students are encouraged to apply the acquired skills acquired through small practical exercises and then through the implementation of a larger project developed throughout the course ..

#### **Evaluation components:**

- Exam: 50%
  - Project: 35%
  - Exercises: 15%
- 

### **Main Bibliography**

Resources provided in: [w3.ualg.pt/~mzacaria/tutorial-uml/index.html](http://w3.ualg.pt/~mzacaria/tutorial-uml/index.html)