

**Software Engineering**  
**Bachelor in Informatics Engineering, 2<sup>nd</sup> Course**

# **Software Engineering**

**Course 2022-2023**

Miguel Sánchez-Puebla  
Eduardo Cibrián  
Jose María Álvarez Rodríguez [Coordinator]

# Overview

- Teachers
  - Jose María Alvarez ([josemaria.Alvarez@uc3m.es](mailto:josemaria.Alvarez@uc3m.es)) – COORDINATOR
  - Miguel Ángel Sánchez Puebla ([masrodri@inf.uc3m.es](mailto:masrodri@inf.uc3m.es))
  - Eduardo Cibrián ([ecibrian@inf.uc3m.es](mailto:ecibrian@inf.uc3m.es))
- **Aula Global (web site)**

*“A practical course on software engineering (analysis and design)”*

# The course in context of other degree courses (SEBoK)

Course	PDD	IS	DPDS	TADS	DSIC	MDV
Type (semester)	OB-2-2	OB-3-1	OB-3-2	ES-3-2	ES-4-1	ES-4-1
Software Requirements	i	p	u	u	u	u
Software Design	i	p	u	u	u	p
Software Construction	p			p	p	p
Software Testing	p	u	u	p		p
Software Maintenance			p	p		
Software Configuration Management			p	p		
Software Engineering Management			p	p	p	
Software Engineering Process	i	i	p			
Software Engineering Models and Methods		p	u	u		p
Software Quality			p		p	u
Software Engineering Professional Practice	i		p			
Software Engineering Economics			p			
Computing Foundations						
Mathematical Foundations						
Engineering Foundations						

i = introduction; p = deepening; u = application & use

# Objectives

- High-level analysis and design of software-based systems:
  - **Requirements that** must be satisfied by the system
  - **Conceptual model** (relevant domain knowledge and information)
  - **Software architecture** (structure and components of the system)
- We will learn to...
  - **Write analysis and design specifications**
  - **Use standards** for project documentations
  - Apply **object-oriented techniques and requirements engineering**
- General competences:
  - Abstraction and complex problem solving
  - Critical and reflective reading
  - Team working
  - Oral presentations
  - Error-based learning

# Syllabus

- **Block I. Requirements engineering**
  - Unit 1. Introduction to requirements engineering
  - Unit 2. Elicitation, description and management of requirements
  - Unit 3. Properties, attributes and organization of requirements
  - Unit 4. Types of requirements
- **Block II. Conceptual modeling with UML**
  - Unit 5. Introduction to conceptual modeling
  - Unit 6. Conceptual modeling: classes and objects
  - Unit 7. Conceptual modeling: associations
  - Unit 8. Conceptual modeling: hierarchies
- **Block III. Architectural modeling with UML**
  - Unit 9. Introduction to architectural modeling
  - Unit 10. Architectural modeling: components
  - Unit 11. Architectural modeling: interfaces
  - Unit 12. Architectural modeling: design by contracts

## Course schedule and contents (tentative)

Week	Reference	Unit and main milestones
1	Thursday, September 8, 2022	Course introduction. Software Engineering Introduction.
2	Thursday, September 15, 2022	Unit 1. Introduction to requirements engineering.
3	Thursday, September 22, 2022	Unit 2. Elicitation, description and management of requirements. <b>Teams created and project proposal.</b>
4	Thursday, September 29, 2022	Unit 3. Properties, attributes and organization of requirements. Unit 4. Types of requirements.
5	Thursday, October 6, 2022	Unit 5. Introduction to conceptual modeling. Unit 6. Conceptual modeling: classes and objects
6*	Thursday, October 13, 2022	<b>Mid-term exam-I: Requirements Engineering.</b>
7*	Thursday, October 20, 2022	Unit 7. Conceptual modeling: associations. <b>1<sup>st</sup> final project delivery. Presentation of the 1<sup>st</sup> final project delivery.</b>
8	Thursday, October 27, 2022	Unit 8. Conceptual modeling: hierarchies.
9	Thursday, November 3, 2022	Unit 9. Introduction to architectural modeling. Unit 10. Architectural modeling: components
10*	Thursday, November 10, 2022	<b>Mid-term exam-II: Conceptual Modelling</b>
11	Thursday, November 17, 2022	Unit 11. Architectural modeling: interface
12	Thursday, November 24, 2022	Unit 12. Architectural modeling: design by contract
13	Thursday, December 1, 2022	<b>Mid-term exam-III: Architectural Modelling.</b> <b>2<sup>nd</sup> final project delivery. Presentation of the 2<sup>nd</sup> final project delivery.</b>
14*	Thursday, December 8, 2022	<b>Question proposal preparation.</b> Delivery of individual exercises.
15	Thursday, December 15, 2022	

## Practical Lectures

- Team: 4 members
- Two-phase work: requirements + models
- Activities:
  - **Development and documentation of the project following the guidelines**
    - Hours-counting method as a measurement of effort
    - In the beginning of the document
    - Send by email according to the templates for hours management
  - **Mentoring sessions (attendance is not compulsory)**
    - Teams can present their work to receive feedback and advise
  - **Peer-review**
    - Review reports following the guidelines
  - **Oral presentations and project viva**
    - Individual presentations (just a part)
    - Quality and clarity of responses to teachers' questions

## Project statement 2022-2023

- **Brainstorming** for getting project ideas (one per team)
  - Once an idea is selected it will be published in AulaGlobal
- **Building blocks** to develop the project
  - **Requirements** that must be satisfied by the system
  - **Conceptual model** (relevant domain knowledge and information)
  - **Software architecture** (structure and components of the system)
- An essential part of the work lies on the **proper definition** of a **domain vocabulary to define the system.**



## Deliverables

- Please pay attention to **filenames and deadlines**
- **Two draft documents** (v1 and v2)
  - E.g. ProjectSE-M05.doc: team M05, etc.
  - Submit via a task in Aulaglobal
- **Final project (see guidelines):**
  - Final document + presentations + hours
    - Reviews+responses+reviews sent to other teams
  - E.g. ProjectSE-M05.doc + etc.
  - Submit via a task in Aulaglobal
- **Proposal of questions for the final exam.**

## Document Formatting

- Word, Times New Roman 12 or Arial 10, singled-spaced.
  - Two-sided printed
  - Optionally, PDF (read and copy permissions).
- Length (looking for ***quality, no quantity***):
  - Each delivery is very much half part of the final document
  - Final extension: 30 pages (without annexes)
  - Penalty depending on the number of pages.

## Team working and effort

- 45-60 hours/student are expected (and reasonable).
  - 1 hour lecture -> 1 hour of personal work
- Ratio between individual work and team work shall be 4/1 or 3/1.
  - Need of coordination to assign tasks
  - All team members with the same number of hours is suspicious (it is not part of the mark!)

Name	I	T	TOTAL
Ana García	25	35	<b>60</b>
Juan Gómez	25	35	<b>60</b>
Isabel López	25	35	<b>60</b>
Pedro Fernández	25	35	<b>60</b>
<b>TOTAL</b>	<b>100</b>	<b>140</b>	<b>240</b>

**BAD**

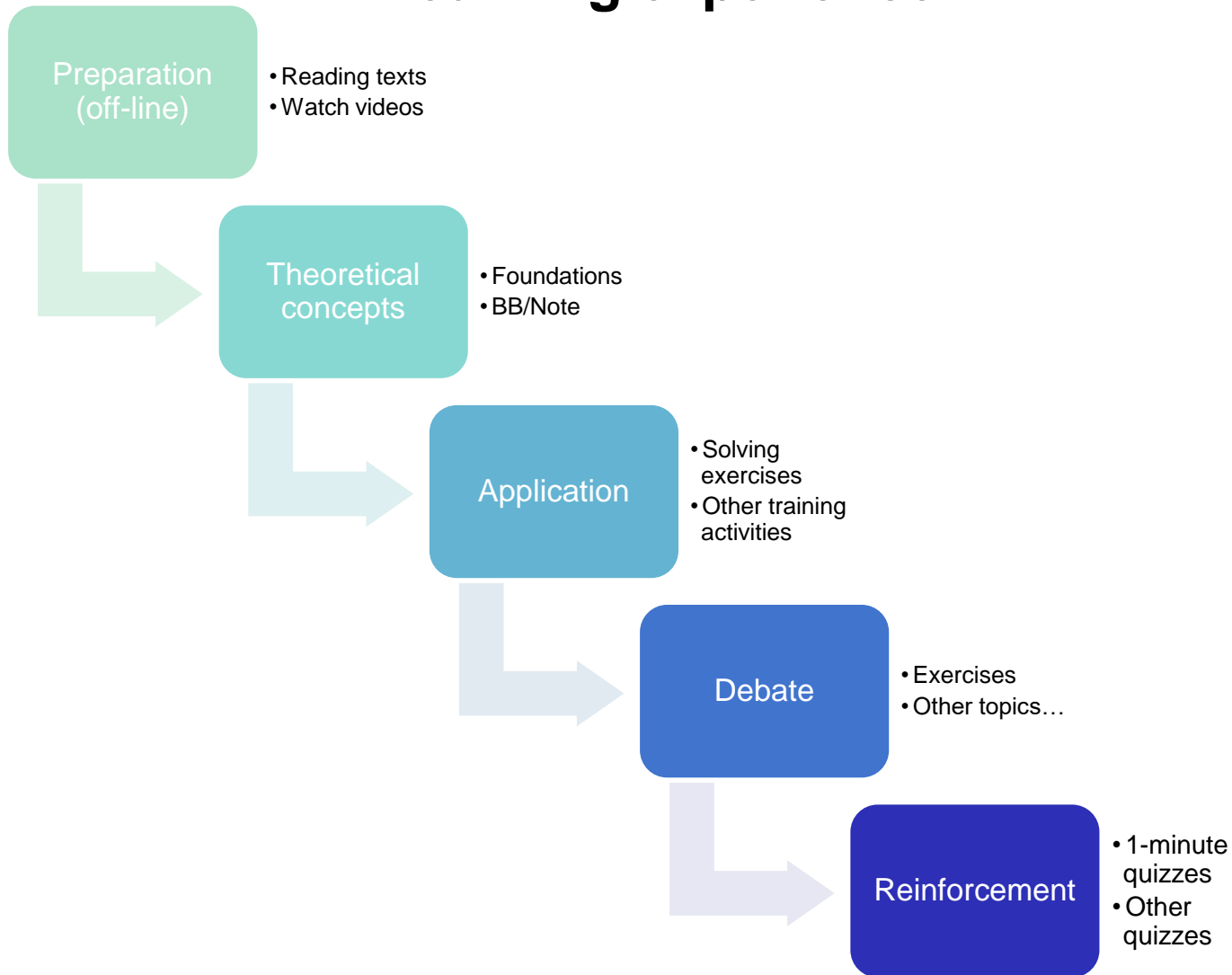
Name	I	T	TOTAL
Ana García	40	15	<b>55</b>
Juan Gómez	43	11	<b>54</b>
Isabel López	47	16	<b>63</b>
Pedro Fernández	50	18	<b>68</b>
<b>TOTAL</b>	<b>180</b>	<b>60</b>	<b>240</b>

**OK**

## Course Activities

- **Theory**
  - Attendance is not compulsory (no attendance control) but VERY recommendable
  - **Open discussion about relevant and hot topics**
  - **Quick questions in AulaGlobal at the end of the lecture/unit**
  - *Lectures are very relevant that's why their weight in the final mark*
- **Lab sessions**
  - Demonstration and learning of tools and exercises
- **Group mentoring**
  - Attendance is not compulsory
  - The teacher can have a real measure of progress (and effort)
  - Take advantage of these sessions to take the most of the teacher. Work hard (before).
- **Presentations/Reviews**
  - Attendance *IS COMPULSORY TO ALL PRESENTATIONS\**.
  - For each team, two members will make the presentation and the others will answer questions
  - Max. time: 10 min/team (presentation + Q&A).
- The official schedule is available at AulaGlobal

# Building a “dynamic” and enriched learning experience



## Assessment System (70% continuous + 30% final-in bold)

	Individual work (50%)	Team work (50%)
Theory (50%)	Mid-term exam-I (10%) Mid-term exam-II (10%) Mid-term exam-III (10%) 1 minute quizzes (10%)	<b>Question proposal (10%)</b>
Practice (50%)	Individual exercises (10%)	1st final project delivery (10%) 2nd final project delivery (10%) <b>Final project delivery (20%)</b>

### Theory is passed *iif*...

- All mid-term exams (mte-I, mte-II, mte-III) are  $\geq 4.5$  and the average (mte-I, mte-II, mte-III) is  $\geq 5$ .
- Otherwise, you will go to the final exam with the pending parts ( $\leq 5$ ).
  - The final exam or a part of the final exam is passed *iif* the grade is  $\geq 5$ .

### Practice is passed *iif*...

- The final project delivery is  $\geq 5$ .
- Otherwise, the final exam will include a part to evaluate the practice.

## Bibliography

- Requirements Engineering
  - Eric Braude. **Software Engineering. An Object-Oriented Perspective**. John Wiley & Sons, 2001.
  - Ian Sommerville. **Ingeniería del Software**. Pearson-Addison Wesley, 2005.
  - Ian Sommerville & Pete Sawyer. **Requirements Engineering: A Good Practice Guide**. John Wiley & Sons. 1997.
  - Roger Pressman. **Ingeniería del software: un enfoque práctico**, 6ª ed. McGraw-Hill.
  - Ian F. Alexander & Richard Stevens. **Writing better requirements**. Addison-Wesley, 2002.
- UML Modelling
  - Martin Fowler, Kendall Scott. **UML Distilled. A Brief Guide to the Standard Object Modeling Language**. Addison-Wesley, 2004.
  - Jim Arlow, Ila Neustadt. **UML and the Unified Process. Practical Object-Oriented Analysis & Design**. Addison-Wesley, 2002.
  - Perdita Stevens, Rob Pooley. **Using UML. Software Engineering with Objects and Components**. Addison-Wesley, 2000.
  - Craig Larman. **Applying UML and Patterns. An Introduction to Object-Oriented Analysis and Design and the Unified Process**. Prentice Hall. 1998.