BASIC DETAILS:

Subject:	FUNDAMENTOS DE INGENIERÍA DEL SOFTWARE		
Id.:	33371		
Programme:	DOBLE GRADO EN FARMACIA Y BIOINFORMÁTICA. PLAN 2018		
Module:	INFORMÁTICA		
Subject type:	OBLIGATORIA		
Year:	3	**Teaching period:**	Primer Cuatrimestre
Credits:	3	**Total hours:**	75
Classroom activities:	27	**Individual study:**	48
Main teaching language:	Inglés	**Secondary teaching language:**	Castellano
Lecturer:	PEREZ PEREZ, MARIA FRANCISCA (T)	**Email:**	mfperez@usj.es

PRESENTATION:

This subject addresses the following contents to be applied during the development of videogames: software development methodologies, agile methods of development, basics for videogame design, and testing strategies.

PROFESSIONAL COMPETENCES ACQUIRED IN THE SUBJECT:

General programme competences	G01	Use learning strategies autonomously for their application in the continuous improvement of professional practice.
	G02	Perform the analysis and synthesis of problems of their professional activity and apply them in similar environments.
	G03	Cooperate to achieve common results through teamwork in a context of integration, collaboration and empowerment of critical discussion.
	G05	Communicate professional topics in Spanish and / or English both orally and in writing.
	G06	Solve complex or unforeseen problems that arise during the professional activity within any type of organisation and adapt to the needs and demands of their professional environment.
	G07	Choose between different complex models of knowledge to solve problems.
	G09	Apply information and communication technologies in the professional field.
	G10	Apply creativity, independence of thought, self-criticism and autonomy in the professional practice.
Specific programme competences	E02	Develop the use and programming of computers, databases and computer programs and their application in bioinformatics.
	E03	Apply the fundamental concepts of mathematics, logic, algorithmics and computational complexity to solve problems specific to bioinformatics.
	E04	Program applications in a robust, correct, and efficient way, choosing the paradigm and the most appropriate programming languages, applying knowledge about basic algorithmic procedures and using the most appropriate types and data structures.
	E05	Implement well-founded applications, previously designed and analysed, in the characteristics of the databases.
	E07	Apply the principles, methodologies and life cycles of software engineering to the development of a project in the field of bioinformatics.
	E08	Evaluate applications and computer systems, previously designed, developed and selected, ensuring their reliability and quality, in accordance with ethical principles and current legislation and regulations.
	E09	Develop and maintain descriptive documentation of the genesis, production and operation of computer systems.
	E10	Design and deploy the architecture of IT systems through the definition of software, hardware and the necessary communications according to some requirements.
	E11	Apply the principles and techniques of concurrent or parallel computing for the creation and simulation of bio-inspired processes.

PRE-REQUISITES:

SUBJECT PROGRAMME:

Subject contents:

1 - Software Process
1.1 - Objectives
1.2 - The Software Process
1.3 - The Software Lifecycle
1.4 - Methodologies
1.5 - Modeling Techniques
1.6 - Case study
2 - Implementation and Software Testing
2.1 - Basic foundations
2.2 - Programming principles and guideliness
3 - Design and redesign
3.1 - Interface design patterns
3.2 - Design patterns
3.3 - Refactorings

Subject planning could be modified due unforeseen circumstances (group performance, availability of resources, changes to academic calendar etc.) and should not, therefore, be considered to be definitive.

TEACHING AND LEARNING METHODOLOGIES AND ACTIVITIES:

Teaching and learning methodologies and activities applied:

This course will use the following methodologies in order to give the students the best opportunity to develop their competences: lectures, practical cases, exercises and cousework presentations.

Participation in class will be accounted in the final score. All readings, practices and works will be announced using the Online University Platform (pdu.usj.es).

Student work load:

Teaching mode	Teaching methods	Estimated hours
Classroom activities	Master classes	12
	Practical exercises	7
	Practical work, exercises, problem-solving etc.	5
	Coursework presentations	1
	Assessment activities	2
Individual study	Tutorials	2
	Individual study	12
	Individual coursework preparation	16
	Group cousework preparation	9
	Research work	5
	Other individual study activities	4
	Total hours:	75

ASSESSMENT SCHEME:

Calculation of final mark:

Written tests:	40	%
Individual coursework:	20	%
Group coursework:	35	%
Participation:	5	%
TOTAL	100	%

*Las observaciones específicas sobre el sistema de evaluación serán comunicadas por escrito a los alumnos al inicio de la materia.

BIBLIOGRAPHY AND DOCUMENTATION:

Basic bibliography:

Sommerville, Ian (2004). Software Engineering, 7th Ed., Pearson.
Pressman, Roger (2005). Software Engineering. A Practitioners Approach, 6th Ed., McGraw-Hill.

Recommended bibliography:

Bjørner, Dines (2006). Software Engineering 3. Domains, Requirements and Software Design, Springer.
Shoval, Peretz (2007). Functional and Object Oriented Analysis and Design: an Integrated Methodology, Idea Group Publishing.
Jalote, Pankaj (2005). An Integrated Approach to Software Engineering, Springer.
McConnell, Steve (2003). Professional Software Development, Addison Wesley
Kroll, Per (2006). Agility and Discipline Made Easy: Practices from OpenUP and RUP, Addison Wesley.
Sangwan, Raghvinder et al (2007). Global Software Development Handbook, Auerbach Publications.
Tomayko, James et al (2004). Human Aspects of Software Development, Charles River Media.
Peckham, Joan (ed.) (2003). Practicing Software Engineering in the 21st Century, IRM Press.
Aurum, Aybüke et al (2005). Engineering and Managing Software Requirements, Springer.
Gunderloy, Mike (2004). Coder to Developer: Tools and Strategies for Delivering Your Software, Sybex
Booch, Grady et al (2007). Object-Oriented Analysis and Design with Applications, 3th Ed., Addison-Wesley.
Pidd, Michael (ed.) (2004). Systems Modelling. Theory and Practice, John Wiley.

Recommended websites:

Center for Systems and Software Engineering: The aim of this site is to work towards evolving and unifying theories and practices of systems and software Engineering.	http://csse.usc.edu/csse/
IEEE Transactions on Software Engineering: Technical articles and news about Software Engineering issues	https://www.computer.org/csdl/trans/ts/index.html
The Podcast for Professional Software Developers: Here you can download audio episodes relating experiences of software engineers	http://www.se-radio.net/