

INSO 4101 - Course Syllabus

1. General Information:

Alpha-numeric codification: INSO 4101
Course Title: Introduction to Software Engineering
Number of credits: 3
Contact Period: 3 hours of lecture per week

Equivalent Course: ICOM 4009

2. Course Description:

English: Introduction to the software development cycle. Models for the software development process and related metrics. Ethical issues in software engineering.

Spanish: Introducción a las actividades del ciclo de desarrollo de software. Modelos de procesos de desarrollo de software y métricas relacionadas. Aspectos éticos en la ingeniería de software.

3. Pre/Co-requisites and other requirements:

Prerequisites: CIIC 4020 or ICOM 4035

4. Course Objectives:

Student will learn how to analyze, design, code, test and document complex software systems. To accomplish this, student will develop a term project, consisting of a substantial application that they will implement using appropriate software technology (frameworks, languages, ...).

5. Instructional Strategies:

- conference discussion computation laboratory
seminar with formal presentation seminar without formal presentation workshop
art workshop practice trip thesis special problems tutoring
research other, please specify:

6. Minimum or Required Resources Available:

Students will use the Departmental computer laboratories to complete course projects.

7. Course time frame and thematic outline

Outline

Contact Hours

Introduction to the course	1
The Software Lifecycle	3
Estimation: Cost, effort and agenda	3
Planning and tracking	3
Risk analysis and management	2
User Interface design	1
UML language	4
Requirements analysis and specification	5
Design principles and concepts, system design testing	6
Software testing	4
Exams, discussion sessions, and presentations	13
Total hours: (equivalent to contact period)	45

8. Grading System

Quantifiable (letters) Not Quantifiable

9. Evaluation Strategies

	Quantity	Percent
<input checked="" type="checkbox"/> Exams	3	30%
<input checked="" type="checkbox"/> Final Exam	1	15%
<input type="checkbox"/> Short Quizzes		
<input type="checkbox"/> Oral Reports		
<input type="checkbox"/> Monographies		
<input type="checkbox"/> Portfolio		
<input checked="" type="checkbox"/> Projects	1	40%
<input type="checkbox"/> Journals		
<input checked="" type="checkbox"/> Other, specify: Homeworks	5-8	15%
TOTAL:		100%

10. Bibliography:

1. Stephen Schach, *Object-Oriented and Classical Software Engineering*, 8th ed., McGraw-Hill, 2010.

2. David Kung, *Object-Oriented Software Engineering: An Agile Unified Methodology*, McGraw-Hill, 2013.
3. Ian Sommerville, *Software Engineering*, 9th ed., Addison-Wesley, 2010.
4. Plfeeger, S.L. and Atlee, J.M., *Software Engineering, Theory and Practice*, 4th Ed., Prentice Hall, 2009.
5. Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1994. [Classic Book]
6. Association for Computing Machinery, *Software Engineering Code of Ethics and Professional Practice*, 1999, <http://www.acm.org/about/se-code>
7. Yingxu Wang, *Software Engineering Foundations: A Software Science Perspective*, CRC Press, 2007. [Classic Book] <http://dx.doi.org/10.1201/9780203496091>. [Available via CRCnetBASE, UPRM General Library Databases]
8. A. Rashid, J. Weckert, and R. Lucas, "Software Engineering Ethics in a Digital World," *Computer*, vol. 42, no. 6, pp. 34–41, 2009. <http://dx.doi.org/10.1109/MC.2009.200> [Available via IEEE Xplore, UPRM General Library Databases]
9. Dines Bjørner, "Software Engineering", Vol. 1-3, Springer, 2006.

11. Course Outcomes

Upon successful completion of this course, the student will be able to:	Program <u>Student Outcomes</u> Impacted
1. understand the engineering nature of software development	2, 6
2. apply principles of engineering to the software development process	2, 6
3. understand the need of effective software testing	2, 6
4. document a software development project through all its phases, complementing informal, semi-formal, and formal means of expression, e.g. technical writing, schematics and diagrams, formal languages	3
5. have a good overview of the tasks, artifacts, skills, knowledge, and trade-offs involved in the phases of software engineering from project inception to software design	1, 2, 6
6. understand the principles of time management as a basis for project management: time tracking, effort estimation, scheduling	6

12. According to Law 51

Students will identify themselves with the Institution and the instructor of the course for purposes of assessment (exams) accommodations. For more information please call the Student with Disabilities Office which is part of the Dean of Students office (Office #4) at (787)265-3862 or (787)832-4040 extensions 3250 or 3258.

13. Academic Integrity

-The University of Puerto Rico promotes the highest standards of academic and scientific integrity. Article 6.2 of the UPR Students General Bylaws (Board of Trustees Certification 13, 2009-2010) states that academic dishonesty includes, but is not limited to: fraudulent actions; obtaining grades or academic degrees by false or fraudulent simulations; copying the whole or part of the academic work of another person; plagiarizing totally or partially the work of another person; copying all or part of another person answers to the questions of an oral or written exam by taking or getting someone else to take the exam on his/her behalf; as well as enabling and facilitating another person to perform the aforementioned behavior. Any of these behaviors will be subject to disciplinary action in accordance with the disciplinary procedure laid down in the UPR Students General Bylaws.—