

University of Puerto Rico  
 Mayagüez Campus  
 College of Engineering  
 Department of Electrical and Computer Engineering  
 Bachelor of Science in Computer Science and Engineering  
 Bachelor of Science in Software Engineering

**Course Syllabus**

<b>1. General Information:</b>	
Alpha-numeric codification: ICOM 5025 Course Title: Object-Oriented Software Development Number of credits: 3 Contact Period: 3 hours of lecture per week Elective in ICOM	
<b>2. Course Description:</b>	
<b>English:</b> Fundamental Concepts of Object-oriented Programming. Analysis, Design, and Development of Object-oriented Software. Study of Object-oriented Languages.	
<b>Spanish:</b> Conceptos Fundamentales de Programación Orientada A Objetos. Analisis, diseño y Desarrollo de Programas Orientados A Objetos. Estudio de Lenguajes Orientados A Objetos.	
<b>3. Pre/Co-requisites and other requirements:</b>	
Pre-requisites: ICOM 4036	
<b>4. Course Objectives:</b>	
Students will learn techniques for object-oriented software design and apply these by completing the design of a software system during the course of the semester.	
<b>5. Instructional Strategies:</b>	
<input checked="" type="checkbox"/> conference <input type="checkbox"/> discussion <input checked="" type="checkbox"/> computation <input type="checkbox"/> laboratory <input type="checkbox"/> seminar with formal presentation <input type="checkbox"/> seminar without formal presentation <input type="checkbox"/> workshop <input type="checkbox"/> art workshop <input type="checkbox"/> practice <input type="checkbox"/> trip <input type="checkbox"/> thesis <input type="checkbox"/> special problems <input type="checkbox"/> tutoring <input type="checkbox"/> research <input type="checkbox"/> other, please specify:	
<b>6. Minimum or Required Resources Available:</b>	
Students will use Departmental computer laboratories to complete course projects.	
<b>7. Course time frame and thematic outline</b>	
<b>Outline</b>	<b>Contact Hours</b>
Introduction	1
Software design process	1
Review of Object-oriented principles	1
Standard design notation	8
Design principles	6
Design patterns and software architectures	4
System design	12
Detailed design	8
Exams and discussions	4

<b>Total hours: (equivalent to contact period)</b>	45

**8. Grading System**

Quantifiable (letters)  Not Quantifiable

**9. Evaluation Strategies** (Suggested): The faculty member teaching the course will provide the student with the evaluation strategy he/she will be using throughout the semester. This will be done within the first week of classes.

	Quantity	Percent
<input checked="" type="checkbox"/> Exams	2	35%
<input checked="" type="checkbox"/> Final Exam	1	25%
<input checked="" type="checkbox"/> Short Quizzes		
<input type="checkbox"/> Oral Reports		
<input type="checkbox"/> Monographs		
<input type="checkbox"/> Portfolio		
<input checked="" type="checkbox"/> Projects	1	40%
<input type="checkbox"/> Journals		
<input type="checkbox"/> Other, specify:		
<b>TOTAL:</b>		<b>100%</b>

**10. Bibliography:**

Partha Kuchana, *Software Architecture Design Patterns in Java*, Auerbach, 2004  
 David Budgen, *Software Design*, 2nd Ed. Addison-Wesley, 2003.  
 Steven John Metsker and William C. Wake, *Design Patterns in Java(TM) (Software Patterns Series)*, Addison-Wesley, 2006.  
 Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1995.

**11. 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 (Chemistry Building, room 019) at (787)265-3862 or (787)832-4040 extensions 3250 or 3258.

Contribution of Course to meeting the requirements of Criterion 5:

Math	Basic Science	General	Engineering Topic
			√

**12. Course Outcomes**

**Map to Program Outcomes**

- |  |     |
|--|-----|
| 1. Describe and analyze the UML and OO model for a given problem.      | (a) |
| 2. Design software from a set of customer requirements.                | (c) |
| 3. Describe and analyze design patterns applicable to problem at hand. | (a) |
| 4. Design modules and test units for software system.                  | (c) |

5. Use an OO language to implement and test software system. (e

Person (s) who prepared this description and date of preparations: Manuel Rodríguez, June, 2008. Submitted by: Manuel Rodríguez, Committee Coordinador, June, 20, 2008.