Metropolitan State University of Denver

Regular Course Syllabus

CS 4250 Software Engineering Principles Spring 2017

Status	completed
Approval Process Name	08. UG Course Modification #2 (Substantive College) (17-18)
Originator	Jody Paul
Created	02/23/2017 05:43PM
Launched	03/06/2017 08:09PM
Tracking:	LAS 1718-27
Department	Mathematical and Computer Sciences, Department of
Prefix:	CS
Course Number:	4250
Course Type:	Computer Science
Course Title:	Software Engineering Principles
Transcript Course Title:	Software Engr Principles
Equivalent/ Crosslisted?	
List all equivalent courses:	
List all crosslisted courses:	
Specify Course Modifications:	description, pre- or co-requisites
Check All That Apply:	Elective
Credit Hours:	4
Schedule Type:	Lecture
Grade Mode:	Letter
Lecture:	60
Lab:	
Internship:	
Practicum:	
Other:	
Additional Student Work Hours per course:	120
Variable topics umbrella course:	No
If yes, number of credits/ repeats allowed	
Specified repeatable course:	No
If yes, number of credits/ repeats allowed	
Prerequisite(s):	COM 2610, CS 3210, CS 3250, MTH 3170, PHI 3370, and 4 additional credits of upper division CS courses, all with a grade of "C" or better, or permission of instructor
Corequisite(s):	N/A
Prerequisite(s) and/or Corequisite(s):	CS 3240 (if corequisite then to be completed in first 8 weeks of semester)

CS 4250

Banner Enforced Prerequisite(s):	COM 2610, CS 3210, CS 3250, MTH 3170, PHI 3370, Additional 4 Upper Division CS credits
Minimum passing grade for Banner enforced prerequisite course(s):	C
Banner Corequisite(s):	
Minimum Passing Grade for Banner enforced corequisite course(s):	
Banner Prerequisite(s) and/or Corequisite(s):	CS 3240
Minimum Passing Grade for Banner enforced pre/corequisite courses:	C
Level	UG
Class	
Program	
Student attribute	
Major	
Other Registration Restrctions	
Catalog Course Description:	This course addresses sound principles and current best practices that support engineering high-quality software. The full life-cycle is considered, from concept inception through product end-of-life. The breadth of stakeholders, their goals, and roles, provides context for decision making and selecting among alternative approaches. Contemporary software engineering techniques, technologies, and tools are explored, reviewed, and applied.
Required Reading and Other Materials will be equivalent to:	Essentials of Software Engineering, Fourth Edition by Frank F. Tsui, et al. Jones & Bartlett (2016); ISBN 1284106004 The Five Dysfunctions of a Team by P. M. Lencioni Jossey-Bass (2002); ISBN 0787960756 The Deadline by Tom DeMarco Dorset House (1997); ISBN 0932633390 The Clean Coder: A Code of Conduct for Professional Programmers by Robert C. Martin Prentice Hall (2011); ISBN 0137081073 Clean Code by Robert C. Martin Prentice Hall (2008); ISBN 0132350882 Debugging Teams: Better Productivity through Collaboration by B. W. Fitzpatrick and B. Collins-Sussman O'Reilly (2015); ISBN 9781491932056 97 Things Every Programmer Should Know: Collective Wisdom from the Experts

	edited by Kevlin Henney O'Reilly (2010); ISBN 0596809484 **+Timely online references and resources**
Specific, Measurable Student Behavioral Learning Objectives:	Upon completion of this course the student should be able to: 1. Categorize principal activities associated with software engineering and relate them to each other 2. Interpret software engineering artifacts 3. Develop software engineering artifacts 4. Assess the effectiveness of a software engineering process and formulate changes to improve the process 5. Demonstrate behaviors beneficial for successful collaborative software engineering 6. Appraise the impact of specific software engineering processes, activities, and artifacts on the quality of outcomes
Detailed Outline of Course Content (Major Topics and Subtopics) or Outline of Field Experience/ Internship	Note that the order of this list is not significant. • Software Life Cycle • Software development life-cycle models: intent and use • Alternative software development life-cycle models • Software Quality • Implications of quality • Quality metrics • Quality assessment • Quality improvement • Human Factors in Software Engineering o Implications of humans engaged in software development o Stakeholder interactions o Humans as software developers o Collaborative development o Organizational factors • Requirements Engineering o Requirements elicitation o Requirements specification o Acceptance testing • Ethical, Legal, Economic, and Social Issues in Software Engineering • Managing Risk in Software Engineering o Risk assessment o Risk mitigation • Software Engineering Technologies and Tools o Computer aids to software engineering o Software engineering best practices
Evaluation of Student Performance	• Participatory contributions • Software engineering artifacts • Written responses • Oral and written presentations • Quizzes/Exams
Learning Objectives	
Distribution of Credit Hours	(4 + 0)

CS 4250

Please Check Box, Confirming That:	The affected departments (found in the impact report) have been notified of these changes. Supporting documentation of this has been attached to this proposal.
Steps	
Originator	
Jody Paul	Approved 03/16/17
Department Curriculum Committee Chair	
Clark Dollard	Approved 03/17/17
Department Chair	
Lindsay Packer	Approved 06/05/17
Dean's Office Tracking Assignment	
Kelsey Smith	Approved 06/06/17
Substantive College Level	
Jody Paul	Approved 06/20/17
Linda Lang-Peralta	Approved 10/09/17
Mona Mocanasu	Approved 09/01/17
Curriculum Manager	
Erica Buckland	Approved 11/14/17
AVP Academic and Student Affairs	
Chad Harris	Approved 11/21/17
Registrar's Office	
Jeremy Coleman	Approved 11/28/17