COLLEGE of CHARLESTON
DEPARTMENT OF COMPUTER SCIENCE

August 2011

Course Description

Computer Science Department, College of Charleston

Course Number: **CSCI 362**
Course Title: **Software Engineering**
Course Coordinator: **Bowring**

Catalog Description (to be updated fall 2012)

This course covers the process of constructing software, including the structural views of software components, and their characteristics and interrelationships, at a high level of abstraction. The course also covers the design principles that govern the purpose, structure, development and evolution of software components. The team projects component of the course uses software design tools to reinforce design processes and associated design representations.
Prerequisite: CSCI 230. Prerequisite or co-requisite: COMM 104

Prerequisites by Topic

OO Programming: Java or C#
Familiar with classes, abstract classes, interfaces, polymorphism
Writing skills for design documentation
Oral communication skills

Major Topics Covered in the Course (Required Topics**)**

1. Documentation: continuous and agile
2. Ethical issues
3. Teamwork
4. Socio-Technical systems
5. Dependability and security
6. Software development processes
7. AgileSoftware Development
8. Requirements Engineering
9. System Modeling
10. Software Testing
11. Software Evolution

Course Narrative

The principal objective of this course is to prepare students for careers as software engineers or software architects by exploring historical and contemporary issues in Software Engineering (SE). These issues include: SE and its relation to computer science and other engineering disciplines, SE licensure and certification, socio-technical systems, safety-critical systems, ethical issues in SE, SE methodologies, development theory, and practice, SE team dynamics, SE project management, SE emerging technologies. Upon completion of this course, students will have a working knowledge of these areas based on extensive readings, research, writing, and speaking assignments. Students will also gain critical skills to analyze and assess SE processes and artifacts and to think holistically about software engineering.

COLLEGE of CHARLESTON
DEPARTMENT OF COMPUTER SCIENCE

August 2011

Course Description

Computer Science Department, College of Charleston

Laboratory projects

Students will work on a term team project to engineer a testing framework for an open-source project that will evolve with the material learned in class. The approach is agile and iterative, so all work is "live" and continuously improved.

Course Outcomes

Upon successful completion of the course, students will be able to:

Course Outcomes	Program Outcome Linkage				
	BS CSCI	BA CSCI	BS INFS	BS DISC	BA CITA
1. Describe the ethical issues associated with software engineering especially with regard to the ACM SE Code of Ethics.	e, g, h				
2. Describe the nature of socio-technical systems illustrated with instances of success/failure resulting from SE practices.	e, g				
3. Work in teams to engineer a testing framework.	a, b, c, d, f, i, j, k				
4. Explain functional, non-functional, and data requirements with attention to the importance of availability, reliability, safety, and security.	b, i				
5. Explain the concepts of software processes and process models with examples.	b, i				
6. Model systems using UML context diagrams, use cases, state diagrams, sequence diagrams, and class diagrams.	a, b, i, j				
7. Describe the purposes and stages of testing and test-driven development.	b, j				
8. Explain the nature of software evolution and the types and cost profiles of maintenance strategies.	g, i, j				
9. Explain the benefits and problems associated with reuse.	g, i, j				
10. Describe the key issues to be considered in designing and implementing distributed software systems including web services and standards, and SOA.	a, b, i, j				
11. Describe the responsibilities of project managers especially re: risks and teams.	c, e, g, h				
12. Describe the basics of project planning, quality management, standards, reviews, and inspections.	b, e, g, i				
13. Explain the basics of change and version management.	b, i				
14. Explain the basics of process improvement.	b, i				
15. Exhibit a basic working knowledge of GUI development using an IDE.	i				
16. Write and present orally analyses of topics in software engineering.	f				

Lasted Edited: 20.August.2011

COLLEGE of CHARLESTON
DEPARTMENT OF COMPUTER SCIENCE

August 2011

Course Description

Computer Science Department, College of Charleston

Oral and Written Communications

Every student is required to submit at least __6_written reports (not including exams, tests, quizzes, or commented programs) of typically _2__ pages and to make _1__oral presentations of typically _5__ minute's duration. Material is graded for grammar, spelling, style, technical content, completeness, and accuracy.

Notes

N/A

Lasted Edited: 20.August.2011