COURSE DESCRIPTION

Dept., Number	CSci 562	Course Title	Software Engineering I
Semester hours	3	Course Coordinator	H. Conrad Cunningham, Professor

Current Catalog Description

Software engineering paradigms, requirement analysis and specifications, design of reliable software; data flow, data structures and object oriented design methodologies.

Note: This course has not been taught in the past six years. This document describes a reformulation of the course with a focus on software design that the instructor is planning for some future offering. He taught the course Engr 660 (Software Engineering II) once with a similar focus.

Textbook

Eric Braude. *Software Design: From Programming to Architecture,* John Wiley and Sons, Inc., 2004. ISBN 0-471-20459-5.

References

Course Outcomes

The goal of this course is to increase the students' abilities to design and construct reliable, flexible, and reusable software, in particular, families of software systems. Upon successful completion of this course, the students:

1. understand the goals and principles of good software design (correctness, robustness, flexibility, reusability, efficiency, and maintainability);
2. know and can apply elements of the Unified Modeling Language (in particular class diagrams and sequence diagrams) appropriately;
3. know several design patterns selected from the standard “Gang-of-Four” creational, structural, and behavioral patterns (e.g., Factory, Abstract Factory, Singleton, Decorator, Composite, Proxy, Visitor, Iterator, Template, and Strategy);
4. can apply design patterns effectively in software design;
5. know the basic concepts of software frameworks;
6. are able to use and design simple software frameworks.

Relationship between Course Outcomes and Program Outcomes

Course outcomes 1, 4, 5, and 6 contribute to program outcomes (c) and (k). Course outcomes 2 and 3 contribute to program outcome (i).

Prerequisites by Topic

Undergraduates registering for the course should have completed a previous course on software engineering (CSci 387).

Major Topics Covered in the Course

1. Information hiding modules; software families (Parnas papers)
2. Software processes
3. Review of object-orientation concepts (using Java)
4. Unified Modeling Language (UML); UML tools
5. Software testing; unit testing with JUnit
6. Design principles
7. Design patterns
8. Software frameworks

Assessment Plan for the Course

This course has not yet been taught with this content. The instructor expects an offering to have approximately 4 examinations and 4-6 homework assignments. Course outcomes 1, 2, 3, and 5 will be assessed primarily using examination questions and secondarily using aspects of the homework assignments. Course outcomes 4 and 6 will likely be assessed primarily using aspects of the homework assignments and secondarily using examination questions. The instructor will evaluate the student performance informally and makes changes to the course content, organization, and pedagogy as appropriate for subsequent offerings of the course

How Data in the Course are Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

Estimate Curriculum Category Content (Semester hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms			Software design		2.5
Data structures			Concepts of programming languages		0.5