CSE 3232 (Approved): Software Requirements Analysis

Course Description

Information systems analysis; object-oriented analysis models and tools; use cases, system modeling using UML; requirements specification development; term project.

Prior Course Number: CSE 616
Transcript Abbreviation: SW Rqmts Analys
Grading Plan: Letter Grade
Course Deliveries: Classroom
Course Levels: Undergrad
Student Ranks: Senior
Course Offerings: Autumn, Spring
Flex Scheduled Course: Never
Course Frequency: Every Year
Course Length: 14 Week
Credits: 3.0
Repeatable: No
Time Distribution: 3.0 hr Lec
Expected out-of-class hours per week: 6.0
Graded Component: Lecture
Credit by Examination: No
Admission Condition: No
Off Campus: Never
Campus Locations: Columbus
Prerequisites and Co-requisites: CSE 3901 or CSE 3902 or CSE 560
Exclusions: Not open to students with credit for CSE 5232 or CSE 616
Cross-Listings:

The course is required for this unit's degrees, majors, and/or minors: No
The course is a GEC: No
The course is an elective (for this or other units) or is a service course for other units: Yes

Subject/CIP Code: 14.0901
Subsidy Level: Baccalaureate Course

Programs

Abbreviation	Description
BS CSE	BS Computer Science and Engineering

Course Goals

Master applying an object-oriented methodology to the analysis of a real-world problem.
Be competent with writing use cases to model functional requirements.
Be competent with using UML use case, class, sequence and collaboration diagrams to model data and behavior requirements.
Be competent with organizational dynamics as it applies to projects.
Be competent with the system lifecycle approach and its phases.
Be familiar with software engineering issues such as correctness, reliability, productivity.
Be familiar with the distiction between analysis and design activities and skills.
Be familiar with working with a team to produce requirements specification document.

Be familiar with the purpose, structure and contents of a requirements specification document.
Be familiar with non-functional requirements such as security, integrity, response time and reliability.
Be familiar with using a UML tool.

Course Topics

Topic	Lec	Rec	Lab	Cli	IS	Sem	FE	Wor
Exams	4.0							
Software engineering, software development life cycle, traditional vs object-oriented analysis	7.0							
Teamwork and organizational dynamics	2.0							
Data flow diagramming (as context model)	4.0							
UML (use case diagram, sequence diagram, collaboration diagram, state chart diagram, class diagram)	15.0							
Requirements elicitation (face-to-face meetings and presentations to real-world client)	8.0							

Representative Assignments

Term project is designed to encourage students to apply principles of Systems Analysis in a Real World context. The project consists of these elements: 1) Students will meet several times with a selected industry client to elicit requirements for a system. 2) Students will prepare and present formally documented system requirements to the client using UML notation for feedback. 3) Students will deliver a detailed and complete Systems Requirement Specification, using traditional and object-oriented notations.
Homework assignments are designed to give students the opportunity to practice analysis concepts in a more controlled framework.

Grades

Aspect	Percent
Homework (4)	20%
Quizzes (2)	20%
Team project (including weekly progress reports, presentation, drafts and final document -- weighted by peer evaluation)	35%
Final Exam	25%

Representative Textbooks and Other Course Materials

Title	Author
An Introduction to Object-Oriented Analysis	Brown

ABET-EAC Criterion 3 Outcomes

Course Contribution		College Outcome
**	a	An ability to apply knowledge of mathematics, science, and engineering.
	b	An ability to design and conduct experiments, as well as to analyze and interpret data.
**	c	An ability to design a system, component, or process to meet desired needs.
***	d	An ability to function on multi-disciplinary teams.
***	e	An ability to identify, formulate, and solve engineering problems.
***	f	An understanding of professional and ethical responsibility.

Course Contribution		College Outcome
***	g	An ability to communicate effectively.
*	h	The broad education necessary to understand the impact of engineering solutions in a global and societal context.
*	i	A recognition of the need for, and an ability to engage in life-long learning.
*	j	A knowledge of contemporary issues.
***	k	An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

BS CSE Program Outcomes

Course Contribution		Program Outcome
	a	an ability to apply knowledge of computing, mathematics including discrete mathematics as well as probability and statistics, science, and engineering;
	b	an ability to design and conduct experiments, as well as to analyze and interpret data;
**	c	an ability to design, implement, and evaluate a software or a software/hardware system, component, or process to meet desired needs within realistic constraints such as memory, runtime efficiency, as well as appropriate constraints related to economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability considerations;
**	d	an ability to function on multi-disciplinary teams;
*	e	an ability to identify, formulate, and solve engineering problems;
**	f	an understanding of professional, ethical, legal, security and social issues and responsibilities;
**	g	an ability to communicate effectively with a range of audiences;
*	h	an ability to analyze the local and global impact of computing on individuals, organizations, and society;
*	i	a recognition of the need for, and an ability to engage in life-long learning and continuing professional development;
**	j	a knowledge of contemporary issues;
***	k	an ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional;
***	l	an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
*	m	an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
**	n	an ability to apply design and development principles in the construction of software systems of varying complexity.

Prepared by: Neelam Soundarajan