

Software Engineering Program

EE379K - Requirements Engineering

Syllabi

Instructor: Dr. K. Suzanne Barber

Textbook: "Managing Software Requirements: A Unified Approach," by Dean Leffingwell and Don Widrig

Office Hours: MW 2:00pm – 3:15pm, ACES 5.436

Course Objective: This course will teach students how to acquire, model, analyze and disseminate requirements to guarantee (1) an accurate depiction of the intent of stakeholders such as end users, experts, and stakeholders responsible for system delivery and (2) sufficient guidance to those system stakeholders responsible for system delivery and maintenance including: system architect, designer, developer, tester, system integrator, and maintenance. Students will learn to construct well-formed models from unstructured input to accurately convey requirements, and synthesize requirements across similar and disparate viewpoints. The course is designed to teach students how to (1) acquire requirements from a multitude of sources, (2) delineate between different types of requirements, (3) select the best model representations to capture those requirements, (4) select the best modeling tools and programming languages to specify computational models of requirements, (5) develop computer-based query functions to analyze modeled requirements, (6) participate and succeed in teams, (7) clearly and concisely present objectives and findings, and (8) professionally deliver reports and presentations.

Course Description:

The course will address six major topics: (1) Planning the Requirements Engineering Effort, (2) Requirements Acquisition Techniques, (3) Requirements Modeling Techniques, (4) Synthesis of Requirements, (5) Analysis of Requirements, and (6) Traceability.

- *Planning the Requirements Acquisition Effort* section focuses on project management issues related to both the strategic and tactical planning required for efficient and productive Requirements Engineering. In this section, students will learn the tactics and techniques for two over-arching strategies: Scenario-based Requirements Acquisition and Model-based Requirements Representation and Analysis.
- The *Requirements Acquisition Techniques* section motivates the utilization of particular acquisition techniques by answering the following question: Requirements are captured to answer WHAT questions? The premise follows: Techniques used to acquire requirements are influenced by what kind of the requirements are important to respective stakeholders (end-users, developers, system architects, program managers, etc.), specifically, what questions stakeholders want results of the Requirements Engineering effort to answer.
- The *Requirements Modeling Techniques* section overviews the different types of modeling techniques from Informal, Semi-formal (i.e. Model-based) to Formal (e.g. Specification Languages). Semi-formal models are promoted. The Semi-formal computational model representations are presented as a vehicle to store requirements and structure the acquired requirements to promote subsequent, insightful analysis. Students will be required to create graphical and computational models. As a design exercise, students will be given the challenge of selecting the best requirements modeling tools and programming languages for representing their requirements models.
- The *Synthesis of Requirements* section will present methods and analysis for merging requirements from different stakeholders and different viewpoints (perspectives).
- The *Analysis of Requirements* section presents Requirements Evaluation in three primary categories: quality, type, stakeholder interest. The Evaluation of Requirements Quality will determine of completeness, consistency, correctness and clarity of the requirements modeled. The Evaluation of Requirements Types will investigate techniques for analyzing and separating requirements by type: business process related requirements and technology related requirements. Separation is viewed as important since these types of requirements change at different rates and pose different kinds of constraints on the development process. Students will learn and implement analysis techniques to interrogate and evaluate computationally modeled requirements. *Traceability*: Considering the iterative process of acquiring, modeling, synthesizing and analyzing, the topic of traceability will be discussed throughout the course.

Course Evaluation:

Students will be evaluated to determine: (1) their grasp of knowledge presented and (2) their ability to apply this knowledge about requirements engineering to problems. Course grades will be assigned as follows:

Homework:	70% (Assignment #1 = 10%; Assignment #2 = 20%; Assignment #3 = 20%; Assignment #4 = 20%)
Participation	10%
Final Exam	20%