

SFWE 301: Software Requirements Analysis and Test

COURSE SYLLABUS

Course Description

This course will teach students how to derive and develop software requirements that are measurable, testable and lead to a compliant software design and implementation. Using industry best practices and tools, students will learn how to elicit, analyze, specify, and validate functional and non-functional software requirements. Students will develop software requirements models and specifications that capture the customer / user's needs. They will also develop test plans and test procedures used in a formal software acceptance test to validate that the developed product meets its requirements as specified. In doing so, students will learn and use basic Java language constructs to implement specified requirements. Additionally, students will also learn how to establish and maintain a software requirement configuration baseline, and the processes used to incorporate subsequent changes, updates, and enhancements to the software requirements over time.

Instructor and Contact Information

Instructor Name: Dr Diana Saldana Jimenez

Email: dianasaldana@arizona.edu

Office: Old Engineering Building Room 266

Office Hours: Mondays and Wednesdays from 2PM to 3PM

Appointments can be made outside of normal office hours by contacting the instructor(s) to schedule a time that is mutually convenient.

You are encouraged to reach out to your instructor frequently throughout the semester via email, phone, text, office hours, or a scheduled synchronous meeting (in-person or Zoom). Every attempt will be made to respond to questions and concerns that you may have within 24 hours.

Course Prerequisites

Advanced standing and successful completion of ECE 275.

Course Format and Teaching Methods

This course is simultaneously offered in three modalities – fully online, in-person, and live online – offering opportunities for students to collaborate across modalities. Regardless of your learning modality, this course is structured around weekly progress as outlined in the course schedule. This course will include a combination of lectures, team activities, experiential learning opportunities, whole-class discussions, and web-based assessments. The course is designed to engage and demonstrate key concepts of the materials covered using collaborative and active learning strategies.

Course Objectives

During this course, you will:

1. Compare and contrast functional software requirements from non-functional software requirements.
2. Elicit, analyze, specify, and validate software requirements for a software product.
3. Describe and exercise software requirements management processes and activities, including establishing a software requirements baseline and evaluating/managing changes to that baseline.

4. Learn basic Java programming language syntax to enable modifications to a Java software program to implement new/modified software requirements.
5. Develop requirements for and implement a Java software product that solves a real-world challenge or problem.
6. Develop test plans and procedures used to verify that the software implementation meets its specified requirements.
7. Execute a software acceptance test verifying the software requirements have been implemented as specified.

Expected Learning Outcomes

Upon completion of this course, you should be able to:

- 1) Derive and trace software requirements from higher level system requirements using a variety of common elicitation techniques such as interviews, workshops, document analysis, prototyping, and other similar strategies used in industry. *[ABET Student Outcome 1]*
- 2) Analyze software requirements for implementation feasibility; ensuring the requirements are quantifiable, verifiable and satisfy the customer's business or user objectives. *[ABET Student Outcome 6]*
- 3) Group software requirements to enable efficient mapping to software architectural elements and ultimately the resulting code base. *[ABET Student Outcome 6]*
- 4) Translate user needs into software requirements via models/diagrams and written specifications suitable for comprehension, review, and implementation. *[ABET Student Outcome 3]*
- 5) Develop acceptance criteria and tests to validate that the developed product meets specified requirements that satisfy customer needs and achieves business objectives. *[ABET Student Outcome 1]*
- 6) Implement and test changes to a Java program as specified by approved engineering change requests (ECR). *[ABET Student Outcome 1]*
- 7) Perform a software acceptance / qualification test to verify that a software product implementation meets its specified software requirements. *[ABET Student Outcome 6]*
- 8) Manage software requirements by establishing a requirements baseline and evaluating/tracking any proposed changes to that baseline. *[ABET Student Outcome 1]*
- 9) Utilize commercially available configuration control tools to implement the configuration management (CM) processes on the established requirements baseline. *[ABET Student Outcome 1]*

Textbooks & Software

Required Textbooks (2)

Requirements Engineering for Software and Systems – 3rd Edition

by Phillip A Laplante

ISBN: 9781138196117

Available electronically through [Inclusive Access](#)

Programming in Java with zyLabs

by zyBooks (link to be provided on D2L to purchase the interactive textbook)

Recommended Textbook

Software Requirements– 3rd Edition

by Karl Wiegars and Joy Beatty

ISBN: 9780735679665

Available electronically through the [University Libraries](#)

Required Software

You will either need to install or create an account for the following software – links and instructions are available in the **Start Here** module of the D2L course site.

- **Integrated Development Environment (IDE)** - You will need a Java IDE of your choice for code development.
- **Jama** – Create an account on the Jama Software’s cloud environment for access to this software requirements management tool.

Assignments and Examinations

Homework

There will be regular homework assignments on the topics covered in class, with approximately 8 homework assignments to be completed on an individual basis (not a team basis). There will also be module-based discussion board prompts that each student is required to participate in and will be graded for.

Team Participation

Team participation will be evaluated by the completion of all assigned Team Participation and Challenge Activities in the zyBooks chapters covered throughout the semester. Team participation will also be evaluated in engagement in the in-class activities that are exercises throughout the semester, attend end of semester Acceptance Test Reviews, along with the completion of any module discussion board prompts that are required.

Comprehensive Exam

There will be one comprehensive exam. The exam will be given as an online (D2L), timed exam, administered by a proctor in the classroom or online, that will be available during the regularly scheduled exam time. Online students will be provided with information on how to access the exam on the specified date in the course calendar. **Note: the instructor will give students ample notice of the format, time, and any resulting stipulations about where and how the exam will be administered.**

Semester Project

Team Component - The semester project is a team-based project. Teams will be formed consisting of 4-5 students. Each team will be given a high-level software product description and a preliminary list of system level requirements. The team will be required to develop a Software Requirements Specification, Test Procedures and conduct an Acceptance Test Review showing how the requirements in the SRS were verified. Teams will also be required to develop a Java prototype with a minimum of 4 use cases documented in the requirements specification and test procedures. The project will culminate in a comprehensive Acceptance Test Review and Demonstration with project stakeholders and the rest of the students in the course.

Individual Components - In addition to the team portions of the project (developing the SRS, test procedures and conducting the ATR), you will be required to write a 2-page individual reflection of your experience working on the team, developing the requirements and test plan for the product the team was given, and also any lessons learned that you personally had working on the project.

Team Participation (part of the Individual Components) - Over the course of working on the semester team project, you will be required to individually submit 2 team evaluations for all deliverables for the semester project. Every team member is expected to contribute equally to the project. If there are team dynamics that are preventing a collaborative working environment, it is best to inform the instructor ahead of time so that adjustments can be made to facilitate effective teaming and communication amongst the team.

Your individual final team project grade will be factored by the average score of all team members' inputs from these evaluations. Failure to submit a team evaluation will result in the loss of 10 points from your personal team semester project score

Grade Distribution, Scale & Policies

The grading distribution for course assignments is as follows:

Written Homework Assignments/Labs:	30%
Class Participation:	15%
Comprehensive Exam:	20%
Semester Project (see total grade distribution below):	35%
Team Charter (5%)	
Software Requirement Specification (25%)	
Team Evaluation #1 (5%)	
Test Procedures and Implementation (30%)	
Acceptance Test Review (25%)	
Personal Reflection (project related) (5%)	
Team Evaluation #2 (5%)	
Total	100%

Rubrics will be posted on D2L for all homework assignments.

Late Work Policy

Homework/ Knowledge Checks/ Projects: All homework, knowledge checks, and elements of the semester project are due at the time that it is specified in the course schedule and/or D2L content pages. **Late homework and projects will not be accepted without prior approval by the instructor and will receive 0 points.**

Exams: A make-up exam may only be given under extraordinary circumstances. The student requesting a make-up exam should contact the instructor well in advance and provide *written* documentation for the reason that he/she will not be able to attend the regularly scheduled exam. It is up to the discretion of the instructor to accept the justification provided by the student.

Instructor Grading & Student Appeals Policy

The instructor will make every attempt to provide timely feedback on all assignments, knowledge checks, exams and projects. In most cases, feedback and grades will be given within 72 hours (excluding weekends) of assigned due dates. Rubrics will be provided for all assignments and team projects.

You can dispute any grade that you receive within two weeks from the time the grade was awarded.

If you feel that you have received an unfair assessment of your performance on any given homework assignment, knowledge check, exam or final project, please reach out to the course instructor either in person, via email or over Zoom as soon as possible. Be prepared to provide substantiated claims for your dispute, including any evidence that would support a re-evaluation of your grade.

Grading Scale

The following scale will be used to award final grades:

A	90-100%
B	80-89%
C	70-79%
D	60-69%
E	less than 60%

Incomplete (I) or Withdrawal (W):

Requests for incomplete (I) or withdrawal (W) must be made in accordance with University policies, which are available at <http://catalog.arizona.edu/policy/grades-and-grading-system#incomplete> and <http://catalog.arizona.edu/policy/grades-and-grading-system#Withdrawal> respectively.

Course Behavior Policy

To foster a positive learning environment, students and instructors have a shared responsibility. We want a safe, welcoming, and inclusive environment where all of us feel comfortable with each other and where we can challenge ourselves to succeed. To that end, our focus is on the tasks at hand and not on extraneous activities (e.g., texting, chatting, reading a newspaper, making phone calls, web surfing, etc.).

University Policies

Links to the following UA policies are available at, <https://academicaffairs.arizona.edu/syllabus-policies>:

- Absence and Class Participation Policies
- Threatening Behavior Policy
- Accessibility and Accommodations Policy
- Code of Academic Integrity
- Nondiscrimination and Anti-Harassment Policy
- Subject to Change Statement

Use of Artificial Intelligence (AI) Policy

The University of Arizona Code of Academic Integrity prohibits “all forms of academic dishonesty, including...cheating, fabrication, facilitating academic dishonesty, and plagiarism.” I expect that all work students submit for this course will be *their or their team’s* effort, and an authentic representation of *their or their team’s* knowledge and skills. However, this course encourages students to explore other resources to understand and practice the course material. The use of generative Artificial Intelligence (AI) tools such as ChatGPT is permitted for all homework, project assignments, and discussions. In alignment with emerging transparency and attribution policies, the following guidelines must be followed ([APA Publishing Policies](#)):

- When a generative Artificial Intelligence (AI) model is used in the drafting of a document or report, the use of AI *must* be disclosed through a citation (see [APA Style Blog: How to cite ChatGPT](#)).
- When AI is cited in Homework and Project Assignments, the authors must include in an appendix how, when, to what extent AI was used, and the full output of the conversation. .
- When using AI tools for grammar-checking (i.e. Grammarly), citation generation, or plagiarism detection, citations are *not* necessary.

It is each student's responsibility to assess the validity and applicability of any AI output that is submitted; you bear the final responsibility. Violations of this policy will be considered academic misconduct.