SFWE 504: Software Requirements Analysis and Test
COURSE SYLLABUS

Course Description

In this course you will learn 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, you will learn how to elicit, analyze, specify, and validate functional requirements (what should the software system do) and non-functional software requirements (how should the software system fulfill the functional requirements). You will develop software requirement models and specifications that capture the customer / user's needs. The requirements will be captured in a commercially available software requirements management tool and exported to create a *Software Requirements Specification (SRS)*. You will also develop a formal *Software Test Plan (STP)* that can be used in a software acceptance test to validate that the product will meet its requirements as specified. Additionally, you will also establish and maintain a software requirement configuration baseline, utilizing widely adopted industry processes to incorporate any subsequent additions, deletions, and enhancements to the software requirements over its lifecycle.

Instructor and Contact Information

Instructor Name: Diana Saldana
Email: dianasaldana@arizona.edu
Phone: TBD
Office: TBD
Office Hours: TBD

You are encouraged to reach out to your instructor frequently throughout the semester via email, phone, 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.

Office Hours

Office hours are an opportunity for you to get personalized help with the course material, ask in-depth questions, gain a deeper understanding of the subject or field (both inside and outside the classroom), and connect with your faculty. You are encouraged to come to office hours with specific questions about confusing topics, assessment questions, or feedback, or just to say hello and introduce yourself. I look forward to meeting with you! If you are unable to attend scheduled Office Hours, please email to set up a time that works best for you.

Course Prerequisites

Concurrent enrollment or completion of SFWE 507: Foundations of Software Engineering. In this course, we will build on your ability to:

- outline the phases of the Software Development Lifecycle (SDLC) used in developing, delivering, and maintaining software products for a wide variety of applications.
- explain the relevance of (1) defining software requirements, (2) developing software architectures and designs, and (3) testing, in the development of reliable and resilient software systems.

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 interactions, 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) Capture / document software requirements using a modern software requirements management tool.
4) Exercise software requirements management processes and activities, including establishing a software requirements baseline and evaluating/managing changes to that baseline.
5) Develop a software test plan that can be used as a basis for validating that the product design and implementation will meet its specified requirements.

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.
2) Analyze software requirements for implementation feasibility; ensuring the requirements are quantifiable, verifiable and satisfy the customer's business or user objectives.
3) Categorize and group software requirements that enable effective traceability to software architectural and design elements and ultimately the resulting code base.
4) Translate user needs into software requirements via models/diagrams and written specifications suitable for comprehension, review, and implementation.
5) Develop a *Software Requirements Specification (SRS)* using information captured in a commercially available requirements management tool.
6) Develop acceptance test criteria and a formal *Software Test Plan (STP)* to be used in validating that the developed product will meet the specified requirements to satisfy customer needs and achieve business objectives.
7) Manage software requirements by establishing a software requirements baseline and evaluating/tracking any proposed changes to that baseline.
8) Develop and conduct a formal *Software Requirement Review (SWRR)* of the completed requirements baseline and associated *Software Test Plan*.
9) Collaborate with other graduate-level students in the course to demonstrate effective teamwork throughout the semester project.

Textbooks & Software

All textbooks are available electronically for **free** through the University of Arizona Libraries. Instructions and links for accessing the textbooks are available in D2L.

Required Textbooks

Software Requirements (3rd Edition)
by Karl Wiegars and Joy Beatty
ISBN: 978-0735679665

Recommended Textbooks

Requirements Engineering for Software and Systems (4th Edition)
by Phillip A Laplante
ISBN: 978-1032275994

Required Software

You will need to create an account with **Jama software**, which is a commonly used tool in industry for requirements management. Be sure to use the links and instructions to the university's education license available in D2L; this will give you free access to the tool.

Assignments and Examinations

Individual Homework (x3)

To help you practice introductory course concepts, three individual homework are assigned in Modules 1-3. These homework assignments are based on short software system case studies and consist of several questions or tasks. The questions and case studies are delivered as a single Google Doc that you can use as a template for your typed responses. All homework assignments are submitted as a .doc(x) or .pdf through the D2L Assignments tool. Feedback is provided via a rubric.

Discussions (x7)

Participating in online discussions allows you to solidify your understanding of course topics and explore differing perspectives. There is one discussion prompt included in each module for Modules 2-6. Module 1 will have two discussions - one self-introduction and a second discussion that is content-based. Full participation in discussions includes (1) an initial original post and (2) at least one response post before the end of the module. Your participation in the discussion will be assessed using the discussion rubric posted to D2L. All discussions take place via the D2L Discussion tool.

Comprehensive Exam (x1)

To assess your achievement of course outcomes 1-4, there is one timed (75 min) open note and open book comprehensive exam. In-person students will take the exam in class during the designated class period. Online students have a 24-hour period to access and complete the exam. Specific exam dates are published in the course schedule. You will have one attempt to complete the exam via the D2L Quiz tool (in-person students should bring a laptop to the exam period).

Semester Project

The semester project is an opportunity for you to apply course topics to a real-world software design challenge in a team-based setting (3-4 students/team). The primary deliverables for the project include:

- a Software Requirements Specification (SRS);
- a Software Test Plan;
- an Acceptance Test Review;
- a Prototype (i.e. source code) for two use cases documented in the SRS and Test Plan;
- a Software Review Document

Milestone deliverables, including draft documents, will be periodically collected for feedback ahead of the final submission of project deliverables in Module 7. The semester project will be assessed based on (1) submitted group work, (2) individual contribution, and (3) final presentation.

Group Work (x8)

All project deliverables and milestones will be completed and submitted as a group using the D2L Assignments Tool. All team members will receive the same score on group work; feedback for submitted group work will be given via a D2L Rubric.

Individual Contribution (x3)

Throughout the project, you will be asked to individually complete two group member evaluations and one personal reflection. **Individual evaluations** will be completed through FeedbackFruits' Group Evaluation Tool which will allow you to (1) provide feedback on the contributions of your team members, (2) review your feedback, and (3) reflect on the feedback you received. **One individual written reflection** will be submitted at the conclusion of the project, and provides an opportunity for you to reflect on your experience working on a team, developing the requirements and test plan, and your lessons learned.

Note: If there are team dynamics that are preventing a collaborative working environment, please inform the instructor ahead of time so that adjustments can be made to facilitate effective teaming and communication.

Software Requirements Review Presentation (x1)

The Software Requirements Review presentation is when we share our work, discover diverse perspectives, and gather feedback. To fully engage with this process, we expect everyone to attend and present synchronously - in-person students from the classroom, and online and distance students via Zoom. Presentations are scheduled for the Final Exam date and published in the course schedule. If you are unable to attend and participate as expected for your course modality (i.e. in-person, online, or distance), please email your instructor to further discuss the schedule.

Grade Distribution, Scale & Policies

The grading distribution for course assignments is as follows:

Individual Homework (x3)	15%
Discussions (x7)	10%
Comprehensive Exam (x1)	10%
Semester Project: Group Work (x8)	35%
Semester Project: Individual Contribution (x3)	15%
Semester Project: Presentation (x1)	15%
Total	**100%**

Late Work Policy

Homework, Discussions, and Semester Project: All homework, discussions, and elements of the semester project are due at the time specified in the course schedule or D2L. Late homework, discussions, and project components *will not* be accepted without prior approval by the instructor. You are always encouraged to contact the instructor to discuss your needs for an assignment extension. Submission of a late assignment without prior approval will receive 0 points.

Exams: A make-up exam may only be given under extraordinary circumstances (i.e. medical appointments or family emergencies). 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.

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.

Instructor Grading & Student Appeals Policy

The instructor will make every attempt to provide timely feedback on all assignments, discussions, exams and project components. In most cases, feedback and grades will be given within a week of assigned due dates. Rubrics will be provided for all assignments.

You can dispute any grade that you receive within one week 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, discussion, exam or project component, 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