

**COSC 4336 Software Development, Fall 2022**

**General Course Information**

<b>Instructor</b>	Dr. Yi Li Email: <a href="mailto:yli@uttyler.edu">yli@uttyler.edu</a>
<b>Lecture</b>	M/W/F 9:05 AM - 10:00 AM @ COB 255
<b>Office Hour</b>	M/W 10:00 AM - 11:30 AM @ COB 315.08
<b>Required Text</b>	<a href="#"><i>Systems Analysis and Design</i></a> , by Harry J. Rosenblatt (11 th edition). ISBN-10: 1305494601 ISBN-13: 9781305494602
<b>Suggested Materials</b>	<p><i>Software Engineering: A Practitioner's Approach</i>, by Roger Pressman (9th edition). ISBN-10: 0073375977 ISBN-13: 978-0073375977</p> <p><i>Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development</i>, by Craig Larman(3 rd edition). ISBN-13: 978-0131489066 ISBN-10: 0131489062</p> <p><i>Object-Oriented and Classical Software Engineering</i>, by Stephen R Schach (8 th Edition). ISBN-13: 978-0073376189 ISBN-10: 0073376183</p>
<b>Other Recommendations</b>	<p>Some online magazines:</p> <p><a href="#">Application Development Trends</a></p> <p><a href="#">Application Developer Magazine</a></p> <p><a href="#">Information Week</a></p>
<b>Prerequisites</b>	COSC 2336
<b>Course Description</b>	It presents a step-by-step methodology - that integrates Planning, Requirements Modeling, UML Tools, Interface and Data Design, CASE tools, Implementation, Test-Driven Development, Quality Assurance, Configuration Management, and Agile Principles throughout the life cycle of software development. Students will be assigned to a group project and will work together through the full development cycle, from understanding the requirements to delivering a functioning product, and will make a series of presentations and reports of the work.
<b>Learning Outcomes</b>	<ol style="list-style-type: none"> <li>1. Describe software development methods and life cycle models</li> <li>2. Analyze software project requirements</li> <li>3. Translate the analysis model into the design model</li> <li>4. Describe agile software development</li> <li>5. Describe &amp; implement design concepts</li> </ol>

	6. Introduce state-of-the-art tools for large-scale software development 7. Develop students' ability to evaluate graphical user interfaces 8. Compare testing strategies for unit and integration testing 9. Implement the major software development techniques in practical projects		
<b>Grading Policy</b>	Midterm Exam – 20%	90.0 - 100%	A
	Final Exam – 20%	80.0 - 89.99%	B
	Assignments/Quizzes – 15%	70.0 - 79.99%	C
	Project – 45%	60.0 - 69.99%	D
		Below 60%	F

### Projects

- There will be one semester long project, to be completed in teams.
- In addition to the programming project, each team will give in-class presentations demonstrating the project progress.
- All code and files required for the projects must be submitted via Canvas.
- Peer evaluation will be conducted to track each team member's performance.
- Group meeting logs will be collected weekly to keep track of each team's project progress.
- Team members who do not contribute appropriately to an assignment will receive a significantly lower grade for that assignment than the rest of that team, possibly "zero", at the discretion of the instructor.
- If there is a lack of appropriate contribution on any two or more group assignments, the non-participating student(s) may be recommended to withdraw from the course.
- Project Assignments should be turned in no later than the deadline. Turn in what is completed by the deadline for partial credit. **No late submissions will be accepted.**

### Quizzes

- Each course module includes at least one formative short multiple choice quiz. At the end of each module, students take a summative multiple choice quiz that assesses their knowledge of the concepts covered in the module.

### Assignments

- You have to turn assignments in on time. Points will be deducted for late assignments. A penalty of 30% will be deducted from your score for the first 24-hour period your assignment is late. A penalty of 70% will be deducted from your score for  $\geq$  24-hour period. No credit for  $\geq$  3 days. Weekend days will be counted.

### **Exams**

- All exams must be taken on the day and time scheduled; there will be no makeups for missed exams – all missed exams get a grade of zero.

### **Course Policies**

- You are expected to do your own work. You may assist each other with general concepts, but direct assistance with a particular assignment or any attempts to gain an unfair academic advantage will not be tolerated. Any indication of cheating and/or plagiarism on an exam/assignment/project will be an automatic 0 (zero) for the exam/assignment/project for all students involved. Solutions copied from the internet, instructor's manual, etc. will also be given zero credit. If you have questions about the line between assistance and cheating, discuss it with the instructor. For examples of Scholastic Dishonesty, please visit Section 8-802 of the [Manual of Policy and Procedures](#).

### **Attendance and Participation**

- Attendance and participation will be considered in a portion of the student's grade.
- Each student is expected to participate by making regular forum posts on the discussion board under Canvas, either asking a question or responding to an existing topic.
- Regular course attendance is mandatory. If attendance is low, the instructor reserves the right to administer pop quizzes for credit, to be determined