

Course #:	CSCI-B 565
Course Title:	Software Engineering I
Course Type:	Graduate / Upper level elective for undergraduates
Prerequisites:	C308 Systems Analysis and Design, C442 Database Systems
Credits:	3
Text Book:	Software Engineering by Ian Sommerville
References:	<p>Class notes ACM and IEEE CS</p> <ul style="list-style-type: none"> • Software Engineering Code of Ethics and Professional Practice <p>ACM</p> <ul style="list-style-type: none"> • ACM Code of Ethics
Current Catalog Description:	Analysis, design and implementation of software systems. Requirements specification: data and process modeling. Software design methodologies. Software quality assurance: testing and verification. Software development processes.
Course Goals	<p>The student who completes this course:</p> <ol style="list-style-type: none"> 1. Will be introduced to the basic concepts of software development: system analysis, requirement analysis, specification, design, implementation, testing and maintenance. 2. Will learn how to produce documentation that should accompany large software products. 3. Will learn about working in teams and presentation skills. 4. Will be introduced to advanced topics in Software Engineering.
Major Topics Covered in the Course	<ol style="list-style-type: none"> 1. Software process 2. Project management 3. Critical system analysis, design, implementation and testing 4. Software reuse 5. Software evolution 6. Configuration management 7. Software testing
Laboratory projects (specify number of weeks on each)	<p>One course project is assigned to teams. During these 13 weeks, students need to produce four documents and give four presentations. These documents are:</p> <ul style="list-style-type: none"> • Project proposal (2 weeks) • Project analysis (3 weeks) • Project design (3 weeks)

	<ul style="list-style-type: none"> • Test plan (5 weeks) 																		
Estimate Curriculum Category Content (Semester hours)	<table border="1"> <thead> <tr> <th>Area</th> <th>Core</th> <th>Advanced</th> </tr> </thead> <tbody> <tr> <td>Algorithms</td> <td></td> <td></td> </tr> <tr> <td>Software Design</td> <td>30</td> <td>23</td> </tr> <tr> <td>Comp. Arch.</td> <td></td> <td></td> </tr> <tr> <td>Data Structures</td> <td></td> <td></td> </tr> <tr> <td>Prog. Languages</td> <td></td> <td></td> </tr> </tbody> </table>	Area	Core	Advanced	Algorithms			Software Design	30	23	Comp. Arch.			Data Structures			Prog. Languages		
Area	Core	Advanced																	
Algorithms																			
Software Design	30	23																	
Comp. Arch.																			
Data Structures																			
Prog. Languages																			
Oral and Written Communications	Every student is required to submit at least __4__ written reports (not including exams, tests, quizzes, or commented programs) of typically __6__ pages and to make __4__ oral presentations of typically __20__ minute's duration.																		
Social and Ethical Issues	<p>Approximately one lecture is dedicated to social and ethical issues related to software engineering. Students are provided with online references to ACM and IEEE code of ethics documentation.</p> <ul style="list-style-type: none"> • Software Engineering Code of Ethics and Professional Practice. • ACM Code of Ethics 																		
Theoretical Content	Software process model (CMM)																		
Problem Analysis	The course reviews both classical analysis and object-oriented analysis. We focus on the analysis of critical and complex system.																		
Solution Design	The course reviews architecture design and design pattern. We focus on the design of critical and complex systems.																		
Prepared By	Yu																		