

SWE-211: Software Engineering Syllabus

General Information

Course Number	SWE-211
Credit Hours	3+0 (Theory Credit Hour = 3, Lab Credit Hours = 0)
Prerequisite	None
Course Coordinator	Not Specified

Course Objectives

This course is specially designed for the students who have knowledge about the computers and have good programming skills and want to enhance their level of understanding for software development and analysis. During this course the students will study the software engineering process models, requirements engineering process, and system models. Furthermore, they will learn Methods, tools, notations, and verification and validation techniques for the analysis and specification of software requirements. Moreover, students will be thought about the Introduction to the principles of project management and quality assurances.

Catalog Description

SWE-211

Course Content

Session No.	Weeks	1. Topics	Evaluation Instruments	Suggested Readings
01-02	Week 01	<u>Introduction to Software Engineering</u> 1. General Introduction of Software Engineering 2. FAQs about Software Engineering a. What is software? b. What is software engineering? c. What is the difference between software engineering and computer science? d. What is the difference between software engineering and system engineering? e. What is a software process? f. What is a software process model? g. What are the costs of software engineering? h. What are software engineering methods? i. What are the attributes of good software? j. What are the key challenges facing software engineering?		SOMMERVILL: Chapter 01 PRESMAN: Chapter 01
03-04	Week 01	<u>Introduction to Software Engineering (Cont.)</u>		SOMMERVILL: Chapter 01 PRESMAN:

		<ol style="list-style-type: none"> 1. Software Engineering Diversity <ol style="list-style-type: none"> a. Application Types 2. Software Engineering Fundamentals 3. Software Engineering and the Web <ol style="list-style-type: none"> a. Web Software Engineering b. Web-based Software Engineering 4. Software Engineering Ethics 5. Professional and ethical responsibility 6. Issues of professional responsibility <ol style="list-style-type: none"> a. Confidentiality b. Competence c. Intellectual property rights d. Computer misuse 7. ACM/IEEE Code of Ethics 8. Code of ethics – preamble 9. Code of ethics – principles 10. Ethical dilemmas 11. Key points 		Chapter 01
05-06	Week 02	<u>Introduction to Software Engineering (Cont.)</u> <ol style="list-style-type: none"> 1. Case Study <ol style="list-style-type: none"> a. An Insulin pump control system b. A patient Information System for Mental Health Care c. A wilderness weather station 		SOMMERVILL: Chapter 01 PRESMAN: Chapter 01
07	Week 02	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 01 to 06	
08	Week 02	<u>Software Processes:</u> <ol style="list-style-type: none"> 1. The software process models 2. Software process descriptions 3. Plan-driven and Agile Process 4. Software process models 5. Generic software process models 6. Waterfall model <ol style="list-style-type: none"> a. Waterfall model phases b. Waterfall model problems c. Advantages of the Waterfall model 		SOMMERVILL: Chapter 02 PRESSMAN: Chapter 04
09-10	Week 03	<u>Software Processes (Cont.):</u> <ol style="list-style-type: none"> 1. Incremental development 2. fundamental types of Incremental development <ol style="list-style-type: none"> a. Exploratory development b. Throw-away prototyping 3. Incremental development benefits 4. Problems in Incremental development model 5. Reuse –based software engineering 6. Reuse-oriented development 7. Stages of Reuse-oriented development 8. Advantages of Reuse-oriented development 		SOMMERVILL: Chapter 02 PRESSMAN: Chapter 04

11-12	Week 03	<u>Software Processes (Cont.):</u> 1. Process activities a. Software specification b. Software design and implementation c. Software validation d. Software evolution 2. Requirements engineering process a. Feasibility study b. Requirements elicitation and analysis c. Requirements specification d. Requirements validation 3. Software design and implementation a. Activities in Software design and implementation 4. Software validation a. Stages of Testing b. Testing phases 5. Software Evolution a. System evolution		SOMMERVILL: Chapter 02 PRESSMAN: Chapter 04
13-14	Week 04	<u>Software Processes (Cont.):</u> 1. Coping with Change 2. Reducing the costs of rework 3. Ways to Handle coping with changing system requirements 4. Software prototyping a. Benefits of prototyping b. Prototype development process c. Issues in prototyping		SOMMERVILL: Chapter 02 PRESSMAN: Chapter 04
15-16	Week 04	<u>Software Processes (Cont.):</u> 1. Process iteration 2. Incremental delivery 3. Spiral development 4. The Rational Unified Process a. Static workflows in the Rational Unified Process		SOMMERVILL: Chapter 02 PRESSMAN: Chapter 04
17	Week 05	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 07 to 16	
18	Week 05	<u>Agile Software Development:</u> 1. Introduction 2. Rapid software development 3. Agile methods a. Agile manifesto b. The principles of agile methods c. Agile method applicability 4. Agile development Techniques		SOMMERVILL: Chapter 03
19-20	Week 05	<u>Agile Software Development (Cont.):</u> 1. Extreme programming 2. XP and agile principles 3. The extreme programming release cycle 4. Extreme programming practices 5. XP and agile principles 6. Practical issues in agile methods		SOMMERVILL: Chapter 03

		7. Influential XP practices		
21-22	Week 06			
23-24	Week 06	<u>Agile Software Development (Cont.):</u> <ol style="list-style-type: none"> 1. User stories for requirements 2. Requirements scenarios <ol style="list-style-type: none"> a. A 'prescribing medication' story b. Examples of task cards for prescribing medication 3. Advantages and disadvantages of User stories for requirements 4. Refactoring <ol style="list-style-type: none"> a. Examples of refactoring 5. Testing in XP <ol style="list-style-type: none"> a. Test-first development b. Customer involvement c. Test case description for dose checking d. Test automation e. Problems with test-first development 		SOMMERVILL: Chapter 03
25-28	Week 07	First Mid-Term Exam		
29-30	Week 08	<u>Agile Software Development (Cont.):</u> <ol style="list-style-type: none"> 1. Pair programming <ol style="list-style-type: none"> a. Advantages of pair programming 2. Agile project management 3. Scrum <ol style="list-style-type: none"> a. The Scrum process b. The Sprint cycle c. Teamwork in Scrum d. Scrum benefits e. Distributed Scrum 4. Scaling agile methods <ol style="list-style-type: none"> a. Practical problems with agile methods 5. Contractual issues 6. Agile methods and software maintenance 7. Agile maintenance 8. Agile and plan-driven methods <ol style="list-style-type: none"> a. Agile principles and organizational practice b. Agile and plan-based factors c. System issues d. People and teams e. Organizational issues 9. Agile methods for large systems <ol style="list-style-type: none"> a. Scaling up to large systems b. Multi-team Scrum 10. Agile methods across organization 		SOMMERVILL: Chapter 03
31	Week 08	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 29 to 30	
32	Week 08	<u>Requirement Engineering:</u> <ol style="list-style-type: none"> 1. Requirements engineering 2. What is a requirement? 3. Requirements abstraction 4. Types of requirement <ol style="list-style-type: none"> a. User and system requirements b. Readers of different types of requirements specification 5. Functional and non-functional requirements 6. Agile methods and requirements 7. Functional requirements <ol style="list-style-type: none"> a. Functional requirements for the 		SOMMERVILL: Chapter 04 PRESSMAN: Chapter 10

		Mentcare b. Requirements imprecision c. Requirements completeness and consistency 8. Non-Functional Requirements a. Types of nonfunctional requirement b. Non-functional requirements implementation c. Non-functional classifications d. Examples of nonfunctional requirements in the Mentcare e. Goals and requirements f. Usability requirements g. Metrics for specifying nonfunctional requirements 9. Domain requirements a. Train protection system b. Domain requirements problems		
33-34	Week 09	<u>Requirement Engineering (Cont.):</u> 1. Requirements engineering processes a. A spiral view of the requirements engineering process 2. Requirements elicitation and analysis a. Problems of requirements analysis b. The requirements elicitation and analysis process c. Process activities d. Problems of requirements elicitation 3. Requirements discovery a. Stakeholders in the Mentcare 4. Interviewing a. Interviews in practice 5. Scenarios a. Scenario for collecting medical history in Mentcare 6. Use cases a. Use cases for the Mentcare 7. Ethnography a. Scope of ethnography b. Focused ethnography c. Ethnography and prototyping for requirements analysis		SOMMERVILL: Chapter 04 PRESSMAN: Chapter 10

35-36	Week 09	<u>Requirement Engineering (Cont.):</u> <ol style="list-style-type: none"> 1. Requirements specification <ol style="list-style-type: none"> a. Ways of writing a system requirements specification b. Requirements and design c. Natural language specification d. Guidelines for writing requirements e. Problems with natural language f. Example requirements for the insulin pump software system 2. Structured specifications 3. Form-based specifications <ol style="list-style-type: none"> a. A structured specification of a requirement for an insulin pump 4. Tabular specification <ol style="list-style-type: none"> a. Tabular specification of computation for an insulin pump 5. Use cases <ol style="list-style-type: none"> a. Use cases for the Mentcare system 		SOMMERVILL: Chapter 04 PRESSMAN: Chapter 10
37-38	Week 10			
39-40	Week 10	<u>Requirement Engineering (Cont.):</u> <ol style="list-style-type: none"> 1. The software requirements document <ol style="list-style-type: none"> a. Users of a requirements document b. Requirements document variability c. The structure of a requirements document 2. Requirements Validation <ol style="list-style-type: none"> a. Requirements checking b. Requirements validation techniques c. Requirements reviews d. Review checks 3. Requirements management <ol style="list-style-type: none"> a. Changing requirements b. Requirements evolution c. Requirements management planning d. Requirements change management 		SOMMERVILL: Chapter 04 PRESSMAN: Chapter 10

41	Week 11	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 32 to 40	
42	Week-11	<u>System Modeling:</u> <ol style="list-style-type: none"> 1. Introduction 2. System modeling 3. Existing and planned system models 4. System perspectives 5. UML diagram types 6. Use of graphical models 7. Context models <ol style="list-style-type: none"> a. System Boundaries b. Process perspective c. Process model of involuntary detention 		SOMMERVILL: Chapter 05 PRESSMAN: Chapter 12
43-44	Week 11	<u>System Modeling (Cont.):</u> <ol style="list-style-type: none"> 1. Interaction models <ol style="list-style-type: none"> a. Use case modeling <ol style="list-style-type: none"> i. Transfer-data use case ii. Use cases in the MHC-PMS involving the role 'Medical Receptionist' b. Sequence diagrams <ol style="list-style-type: none"> i. Sequence diagram for View patient information ii. Sequence diagram for Transfer Data 		SOMMERVILL: Chapter 05 PRESSMAN: Chapter 12
45-46	Week 12	<u>System Modeling (Cont.):</u> <ol style="list-style-type: none"> 1. Structural models <ol style="list-style-type: none"> a. Class diagrams <ol style="list-style-type: none"> i. UML classes and association ii. Classes and associations in the MHC-PMS iii. The Consultation class b. Generalization <ol style="list-style-type: none"> i. A generalization hierarchy ii. A generalization hierarchy with added detail c. Object class aggregation models <ol style="list-style-type: none"> i. The aggregation association 2. Behavioral models <ol style="list-style-type: none"> a. Data-driven modeling <ol style="list-style-type: none"> i. An activity model of the insulin pump's operation ii. Order processing b. Event-driven modeling <ol style="list-style-type: none"> i. State machine models ii. State diagram of a microwave oven iii. States and stimuli for the microwave oven iv. Microwave oven operation 		SOMMERVILL: Chapter 05 PRESSMAN: Chapter 12
47	Week 12	<u>System Modeling (Cont.):</u> <ol style="list-style-type: none"> 1. Model-driven engineering <ol style="list-style-type: none"> a. Usage of model-driven engineering b. Model driven architecture c. Types of model <ol style="list-style-type: none"> i. MDA transformations ii. Multiple platform-specific models iii. Agile methods and MDA d. Executable UML 		SOMMERVILL: Chapter 05 PRESSMAN: Chapter 12

		i. Features of executable UML 2. Key Points		
48	Week 12	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 42 to 47	
49-52	Week 13	Second Mid-Term Exam		
53-54	Week 14	<u>Architectural Design:</u> 1. Introduction 2. Software architecture <ol style="list-style-type: none"> Architectural design <ol style="list-style-type: none"> The architecture of a packing robot control system Architectural abstraction Advantages of explicit architecture Architectural representations Box and line diagrams Use of architectural models 3. Architectural design decisions <ol style="list-style-type: none"> Architectural design decisions Architecture reuse Architecture and system characteristics 4. Architectural views <ol style="list-style-type: none"> 4 + 1 view model of software architecture 5. Architectural patterns <ol style="list-style-type: none"> The Model-View-Controller (MVC) pattern The organization of the Model-View-Controller Web application architecture using the MVC pattern 6. Layered Architecture <ol style="list-style-type: none"> The Layered architecture pattern A generic layered architecture The architecture of the LIBSYS system 		SOMMERVILL: Chapter 06 PRESSMAN: Chapter 14
55-56	Week 14	<u>Architectural Design (Cont.):</u> 1. Repository architecture <ol style="list-style-type: none"> The Repository pattern A repository architecture for an IDE 2. Client-server architecture <ol style="list-style-type: none"> The Client–server pattern A client–server architecture for a film library Pipe and filter architecture The pipe and filter pattern <ol style="list-style-type: none"> An example of the pipe and filter architecture 3. Application architectures <ol style="list-style-type: none"> Use of application architectures Examples of application types <ol style="list-style-type: none"> Data processing applications Transaction processing applications Event processing systems Language processing systems Application type examples 4. Transaction processing systems <ol style="list-style-type: none"> The structure of transaction processing applications The software architecture of an ATM system 		SOMMERVILL: Chapter 06 PRESSMAN: Chapter 14

		5. Information systems architecture <ul style="list-style-type: none"> a. Layered information system architecture 6. Language processing systems		
57	Week 15	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 53 to 56	
58	Week 15	<u>Software Testing:</u> <ol style="list-style-type: none"> 1. Introduction 2. Program testing <ul style="list-style-type: none"> a. Program testing goals 3. Validation and defect testing 4. Testing process goals 5. An input-output model of program testing 6. Verification vs validation 7. V & V confidence 8. Inspections and testing <ul style="list-style-type: none"> a. Software inspections b. Advantages of inspections c. A model of the software testing process d. Stages of testing 		SOMMERVILL: Chapter 08 PRESSMAN: Chapter 17, 18
59-60	Week 15	<u>Software Testing (Cont.):</u> <ol style="list-style-type: none"> 1. Development testing <ul style="list-style-type: none"> a. Unit testing <ol style="list-style-type: none"> i. Object class testing ii. The weather station object interface iii. Weather station testing iv. Automated testing v. Automated test components vi. Unit test effectiveness b. Testing strategies <ol style="list-style-type: none"> i. Partition testing <ol style="list-style-type: none"> 1. Equivalence partitioning ii. Testing guidelines (sequences) iii. General testing guidelines 2. Component testing <ul style="list-style-type: none"> a. Interface testing b. Interface errors c. Interface testing guidelines 		SOMMERVILL: Chapter 08 PRESSMAN: Chapter 17, 18
61-62	Week 16	<u>Software Testing (Cont.):</u> <ol style="list-style-type: none"> 1. System testing <ul style="list-style-type: none"> a. System and component testing b. Use-case testing <ul style="list-style-type: none"> i. Collect weather data sequence chart c. Testing policies 2. Test-driven development <ul style="list-style-type: none"> a. TDD process activities b. Benefits of test-driven development c. Regression testing 3. Release testing <ul style="list-style-type: none"> a. Release testing and system testing b. Requirements based testing <ol style="list-style-type: none"> i. Requirements tests c. Features tested by scenario <ol style="list-style-type: none"> i. A usage scenario for the MHC-PMS d. Performance testing 		SOMMERVILL: Chapter 08 PRESSMAN: Chapter 17, 18

		4. User testing <ul style="list-style-type: none"> a. Types of user testing b. The acceptance testing process c. Stages in the acceptance testing process d. Agile methods and acceptance testing 		
63	Week 16	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 58 to 62	
64	Week16	<u>Software Evolution</u> <ul style="list-style-type: none"> 1. Software change 2. Importance of evolution <ul style="list-style-type: none"> a. A spiral model of development and evolution b. Evolution and servicing 3. Evolution processes <ul style="list-style-type: none"> a. Change identification and evolution processes b. The software evolution process c. Change implementation <ul style="list-style-type: none"> i. Urgent change requests ii. The emergency repair process iii. Agile methods and evolution iv. Handover problems 4. Program evolution dynamics <ul style="list-style-type: none"> a. Change is inevitable b. Lehman's laws <ul style="list-style-type: none"> i. Applicability of Lehman's laws 		SOMMERVILL: Chapter 09 (9 th E)
65-66	Week17	<u>Software Evolution (cont.):</u> <ul style="list-style-type: none"> 1. Software maintenance <ul style="list-style-type: none"> a. Types of maintenance b. Maintenance effort distribution c. Maintenance costs d. Development and maintenance costs e. Maintenance cost factors f. Maintenance prediction <ul style="list-style-type: none"> i. Change prediction ii. Complexity metrics iii. Process metrics g. System re-engineering <ul style="list-style-type: none"> i. Advantages of reengineering ii. The reengineering process iii. Reengineering process activities iv. Reengineering approaches v. Reengineering cost factors h. Preventative maintenance by refactoring <ul style="list-style-type: none"> i. Refactoring and reengineering ii. 'Bad smells' in program code 2. Legacy system management <ul style="list-style-type: none"> a. Legacy system categories b. Business value assessment <ul style="list-style-type: none"> i. Issues in business value assessment c. System quality assessment d. Business process assessment e. Factors used in environment assessment f. System measurement 		SOMMERVILL: Chapter 22 PRESSMAN: Chapter 03
67	Week 17	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 64 to 66	
68	Week 17	<u>Dependable systems</u> <ul style="list-style-type: none"> 1. Systems 		SOMMERVILL: Chapter 10

		<ul style="list-style-type: none"> a. Dependability properties b. Sociotechnical systems c. Redundancy and diversity 		
69-70	Week 18	<u>Sociotechnical Systems (Cont.)</u> <ul style="list-style-type: none"> ✧ Dependable processes ✧ Formal methods and dependability 		SOMMERVILL: Chapter 10
71	Week 18	<u>Quick Review of Whole Chapter</u>	Quiz of Sessions 68 to 70	
72-73	Week 19	<u>Project Management:</u> <ol style="list-style-type: none"> 1. Introduction 2. Software project management 3. Software management distinctions 4. Management activities <ul style="list-style-type: none"> a. Proposal writing. b. Project planning and scheduling. c. Project costing. d. Project monitoring and reviews. e. Personnel selection and evaluation. f. Report writing and presentations. 5. Management commonalities 6. Project staffing 7. Project planning 8. Types of project plan 9. Project planning process 10. The project plan 11. Project plan structure <ul style="list-style-type: none"> a. Introduction b. Project organization c. Risk analysis d. Hardware and software resource requirements e. Work breakdown f. Project schedule g. Monitoring and reporting mechanisms 12. Activity organization 13. Milestones in the RE process 14. Project scheduling 15. The project scheduling process 16. Scheduling problems 17. Bar charts and activity networks 18. Task durations and dependencies 19. Activity network 20. Activity timeline 		SOMMERVILL: Chapter 22 PRESSMAN: Chapter 03
74	Week 19	<u>Project Management (cont.):</u> <ol style="list-style-type: none"> 1. Task durations and dependencies 2. Activity network 3. Activity timeline 4. Staff allocation 5. Risk management 6. Software risks 7. The risk management process <ul style="list-style-type: none"> a. Risk identification b. Risk analysis c. Risk planning d. Risk monitoring 8. The risk management process diagram 9. Risk identification 		SOMMERVILL: Chapter 22 PRESSMAN: Chapter 03

		a. Technology risks. b. People risks. c. Organizational risks. d. Requirements risks. e. Estimation risks. 10. Risks and risk types 11. Risk analysis a. Risk analysis (i) b. Risk analysis (ii) 12. Risk planning a. Risk management strategies (i) b. Risk management strategies (ii) 13. Risk monitoring 14. Risk Indicators 15. Key Points		
75-77	Week 19	Final Exam		

Text Book

1. IAN SOMMERVILLE, “Software Engineering”, Tenth Edition

Reference Material

ROGER S. PRESSMAN , “Fundamentals of Software Engineering”, Seventh Edition

Course Learning Outcomes

	Course Learning Outcomes (CLO)
1	Understand the software development process in relation to the fundamental principles and methodologies in software engineering and computer science, including the analysis, design, implementation and testing of contemporary software systems.
2	Apply the software process and methods to real-life software development.

CLO-SO Map

	SO IDs										
CLO ID	a	b	c	d	e	f	g	h	i	j	k
CLO 1	1	0	0	0	0	0	0	0	0	0	0
CLO 2	0	1	1	1	0	0	0	0	0	0	0

Approvals

Prepared By	Irum Sindhu
Approved By	Not Specified
Last Update	16/01/2019