

SYLLABUS: sample

Date / Revision April 2017/September 2017/IT
Faculty Engineering
Study Program Computer Science

SUBJECT: SOFTWARE ENGINEERING 2

1 Basic Information

1.01	Subject Name	SOFTWARE ENGINEERING 2
1.02	Semester	5
1.03	Level	2
1.04	SKS	2
1.05	Mandatory / Curriculum	D-05
1.06	Subject Code	SWEN
1.07	Subject Code	CDSE-D-SWEN-217
1.08	Year	2017
1.09	Quality Control	Final Test, see evaluation
1.10	Limitations	Min 12 and Max 32 students in one class
1.11	Combined with	
1.12	Perquisite	SOFTWARE ENGINEERING 1
1.13	Responsible	
1.14	Revision	September 2017

2 Description of Subject

This course covers provides an extensive hands-on experience in a large scale software product in multi-dimensional problem, which involves technical considerations as well as managerial and organizational considerations, which involves project planning and management, analysis of requirements, construction of software architecture and design, implementation . The course will also provide a live project work.

3 Objectives

The aim of this course is to solve wider systems in which software plays a role. The objective is to expose the student to technical issues about a large software development project, which involves practical and professional issues in software engineering.

4 Competency

- To be able to apply and to implement large scale Software Requirements
- To be able to apply and implement large scale of Software life cycle which include Design, Implementation and Testing

5 Learning Approach / Methodology

- Lectures/ Class contact (time-tabled) supplemented with interactive questions and answers;
- Student Study Effort: homework/assignment; preparation for test/quizzes/ examination.

6 Evaluation

5.1	Absence maximum	25%
5.2	Participation in Discussion	05 Points
5.3	Homework / Classwork	05 Points
5.4	Presentation /Simulation	10 Points
5.5	Daily Quiz	20 Points
5.6	Final Examination	60 Points
	Total	100 Points

7 Text Book and Reference

1	Main Text Book: Software Engineering: A Practitioner's Approach, 7/e. Roger S Pressman, McGraw hill, 2010 ISBN: 978-0-07-337597-7
2	Supplement Textbooks: Software Engineering, 9 th Edition. Ian Sommerville, Addison-Wesley 2011, ISBN-10: 0-13-703515-2

8 Content / Topics of Lecture

Week	Content/Topics of Lecturing	Text Book Chapter	Remark
1	SOFTWARE AND SOFTWARE ENGINEERING: The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths, How It All Starts,	CHAPTER 1	
2	PROCESS MODELS: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process	CHAPTER 2	
3	PRINCIPLES THAT GUIDE PRACTICE: Software Engineering Knowledge, Core Principles, Principles That Guide Each Framework Activity	CHAPTER 4	
4	UNDERSTANDING REQUIREMENTS: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements	CHAPTER 5	
5	DESIGN CONCEPTS: Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model	CHAPTER 8	
6	PATTERN-BASED DESIGN: Design Patterns, Pattern-Based Software Design, Architectural Patterns, Component-Level Design Patterns, User Interface Design Patterns, WebApp Design Patterns	CHAPTER 12	
7	QUALITY CONCEPTS: What Is Quality?, Software Quality, The Software Quality Dilemma, Achieving Software Quality	CHAPTER 14	
8	Semester Break		
9	SOFTWARE QUALITY ASSURANCE: Background Issues, Elements of Software Quality Assurance, SQA Tasks, Goals, and Metrics, Formal Approaches to SQA, Statistical Software Quality Assurance, Software Reliability, The ISO 9000 Quality Standards, The SQA Plan	CHAPTER 16	
10	SOFTWARE TESTING STRATEGIES: A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, The Art of Debugging	CHAPTER 17	

11	FORMAL MODELING AND VERIFICATION: The Cleanroom Strategy, Functional Specification, Cleanroom Design, Cleanroom Testing, Formal Methods Concepts, Applying Mathematical Notation for Formal Specification, Formal Specification Languages	CHAPTER 21	
12	PRODUCT METRICS: A Framework for Product Metrics, Metrics for the Requirements Model, Metrics for the Design Model, Design Metrics for WebApps, Metrics for Source Code, Metrics for Testing, Metrics for Maintenance	CHAPTER 23	
13	PROJECT MANAGEMENT CONCEPTS: The Management Spectrum, People, The Product, The Process, The Project, The W5HH Principle, Critical Practices	CHAPTER 24	
14	RISK MANAGEMENT: Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring, and Management, The RMMM Plan	CHAPTER 28	
15			