

KOCW Content Development Application

	Faculty: Endicott College of International Studies Department & Major: AI & Big Data Department		
Name	Dr. Hasan TINMAZ	Position	Assist. Prof. Dr.
Contact		E-mail	
Subject	Introduction to Software Engineering		
Credit	3 Credits		
Field	Liberal Arts () Social Science () Engineering (X) Natural Science () Education () Medicine or Pharmaceutical Study () PE or Art ()		
Outline of the Class	This course introduces software engineering principles, including discussions of development methodologies, requirements analysis, project planning, software design, software construction, software management, software quality, and software testing. The students will gain experience in the life-cycle development of software systems.		
Weeks	3 Credits * 15 minutes * 10 weeks = 450 minutes (10) Weeks (1 credit should be more than 15 minutes of a video each week for 10 weeks.)		
I submit this document for the KOCW Development Project. <div style="text-align: right; margin-top: 10px;">  </div>			
2022.04.29. * Contact: Tel. 042-630-9396, 9285 / WCTL3@wsu.ac.kr			

KOCW Content Development Application

1. Outline

가. Name of a Class

Class Name	Introduction to Software Engineering		
Semester	Second semester of 2022	Division	Major (X) Liberal Art ()

나. Goal of a Class

(1) **Goal of a Class:** Upon completion of this course, students will be able to:

- understand software engineering concepts, principles, and techniques,
- understand the phases and activities of the software process,
- have gained experience in practical software development work within the framework of integrated development environments

(2) **Introduction:** This course introduces software engineering principles, including discussions of development methodologies, requirements analysis, project planning, software design, software construction, software management, software quality, and software testing. The students will gain experience in the life-cycle development of software systems.

2. Weekly Plan

Week	Content (Topic)	Learning Objective	How to Operate		
			Methodology	Material	Reference
1	Introduction to Software Engineering	At the end of this lecture, the students will understand what software engineering is and why it is important and understand that the development of different types of software system may require different software engineering techniques.	Lecture and discussion of broad themes in software engineering.	PPT	Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.
2	Software Processes I	At the end of this lecture, the students will understand the concepts of software processes and software process models and be introduced to three general software process models and when they might be used.	Lecture and discussion of broad themes in software processes.	PPT	Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.
3	Software Processes II	At the end of this lecture, the students will know about the fundamental process activities of software requirements engineering, software development, testing, and evolution; understand why processes should be organized to cope with changes in the software requirements and design; and understand the notion of software process improvement and the factors that affect software process quality.	Lecture and discussion of broad themes in software processes.	PPT	Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.
4	Agile Software Development I	At the end of this lecture, the students will understand the rationale for agile software development methods, the agile	Lecture and discussion of broad themes in agile software	PPT	Software Engineering (2016, 10th ed.) Global Edition.

Week	Content (Topic)	Learning Objective	How to Operate		
			Methodology	Material	Reference
		manifesto, and the differences between agile and plan-driven development; and know about important agile development practices such as user stories, refactoring, pair programming and test-first development.	development.		<i>Ian Sommerville.</i> Pearson.
5	Agile Software Development II	At the end of this lecture, the students will understand the Scrum approach to agile project management and understand the issues of scaling agile development methods and combining agile approaches with plan-driven approaches in the development of large software systems.	Lecture and discussion of broad themes in agile software development.	PPT	Software Engineering (2016, 10th ed.) Global Edition. <i>Ian Sommerville.</i> Pearson.
6	Requirements Engineering	At the end of this lecture, the students will understand the concepts of user and system requirements and why these requirements should be written in different ways; understand the differences between functional and non-functional software requirements; understand the main requirements engineering activities of elicitation, analysis, and validation, and the relationships between these activities; and understand why requirements management is necessary and how it supports other requirements engineering activities.	Lecture and discussion of broad themes in requirements engineering.	PPT	Software Engineering (2016, 10th ed.) Global Edition. <i>Ian Sommerville.</i> Pearson.

Week	Content (Topic)	Learning Objective	How to Operate		
			Methodology	Material	Reference
7	System Modeling	At the end of this lecture, the students will understand how graphical models can be used to represent software systems and why several types of model are needed to fully represent a system; understand the fundamental system modeling perspectives of context, interaction, structure, and behavior and understand the principal diagram types in the Unified Modeling Language (UML) and how these diagrams may be used in system modeling.	Lecture and discussion of broad themes in system modeling.	PPT	Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.
8	Design and Implementation	At the end of this lecture, the students will understand the most important activities in a general, object-oriented design process; understand some of the different models that may be used to document an object-oriented design and know about the idea of design patterns and how these are a way of reusing design knowledge and experience.	Lecture and discussion of broad themes in software design and implementation.	PPT	Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.
9	Software Testing	At the end of this lecture, the students will understand the stages of testing from testing during development to acceptance testing by system customers; have been introduced to techniques that help you choose test cases that are geared to discovering program defects;	Lecture and discussion of broad themes in software testing.	PPT	Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.

Week	Content (Topic)	Learning Objective	How to Operate		
			Methodology	Material	Reference
		<p>understand test-first development, where you design tests before writing code and run these tests automatically; know about three distinct types of testing—component testing, system testing, and release testing; and understand the distinctions between development testing and user testing.</p>			
10	Software Evolution	<p>At the end of this lecture, the students will understand that software systems have to adapt and evolve if they are to remain useful and that software change and evolution should be considered as an integral part of software engineering; understand what is meant by legacy systems and why these systems are important to businesses; understand how legacy systems can be assessed to decide whether they should be scrapped, maintained, reengineered, or replaced; have learned about different types of software maintenance and the factors that affect the costs of making changes to legacy software systems.</p>	<p>Lecture and discussion of broad themes in software evolution.</p>	PPT	<p>Software Engineering (2016, 10th ed.). Global Edition. <i>Ian Sommerville</i> Pearson.</p>

※ You can freely complete the content sections based on the feature of the class.

3. How are you going to use your class?

For a credit class (), For a non-credit class (), For a public view ()

4. Expected Outcome

There is an ongoing demand on the production of software in parallel to the recent advancements in information and communication technologies. Therefore, software engineering plays a critical role for our future. We must continue educating software related engineers and develop the discipline so that we could meet the demand for more software / applications and create the increasingly complex future systems that we need.

The course is principally aimed at university students taking introductory and advanced courses in software and systems engineering. Moreover, the software engineers in industry may find the course useful in general to update their knowledge on topics such as software reuse, architectural design, security and systems engineering.