

Software Special Development 1

COURSE SYLLABUS

Spring Semester 2008

BASIC INFORMATION

Instructor: JUNBEOM YOO
Office: New Millennium Bldg. Room 904
Office Phone: 02-450-3258
E-Mail: jbyoo@konkuk.ac.kr
Homepage: <http://home.konkuk.ac.kr/~jbyoo>
Course Page: <http://home.konkuk.ac.kr/~jbyoo/08SSD1>
Class Hours: 09:00 ~ 11:00 (Wednesday), 13:00 ~ 15:00 (Friday)

DESCRIPTION

This course involves a complete development of application software on the basis of theory, methodology and tools experienced from previous courses. Emphases are on step-by-step development producing a set of official documents as well as executable codes. It also introduces students to two practical case studies. This course assumes that students are familiar with fundamentals of software engineering, UML and programming languages, i.e. C, C++, Java, etc.

COURSE ORGANIZATION

This course is a lecture-lab course in which topics are presented by the instructor, and assigned practices are completed by students during the lab periods. Each group of 4 students performs a team project, and presents its progress four times. (Planning, Analysis, Design and Final)

COURSE OBJECTIVE

1. To remind students of the fundamentals of software engineering.
2. To introduce students to a practical example of software development process.
3. To provide students with opportunities to develop a full set of software in accordance with industrial standards.

COURSE TOPICS

1. Fundamentals of Software Engineering

2. Object-Oriented Analysis and Design (OOA / OOD)
3. Rational Unified Process (RUP)
4. Object Space Process (OSP)
5. Case Studies: Library Management System
6. Case Studies: Flight Reservation System
7. Actual development of application software
8. Introduction to Formal Methods

TEXT

1. Required Text: "Lecture Note: Software System Development", by Hanium (<http://hanium.or.kr>)
2. Auxiliary Text:
 - A. *Applying UML and Patterns*, by Craig Larman
 - B. *Software Engineering: A Practitioner's Approach*, by Roger S. Pressman
 - C. *The Rational Unified Process Made Easy*, by Grady Booch
 - D. *System and Software Verification*, by B.Bérard, et. al

GRADING PLAN

1. Attendance 10%
2. Mid-term Exam. 20%
3. Final Exam. 30%
4. Practice 10%
5. Team Project 30%

TENTATIVE SCHEDULE

WEEKS	DATE	PHASE	TOPICS / ACTIVITY	PRACTICE
1	03.01 ~ 03.07	Introduction	Fundamentals of Software Engineering Software Process Model	#1
2	03.10 ~ 03.14	Planning (Theory)	OSP Stage 1000 – Plan and Elaboration	#2
3	03.17 ~ 03.21	Planning (Case Study)	Flight Reservation System / Library Management System	#3
4	03.24 ~ 03.28	Planning (Discussion)	Team Project Presentation #1	
5	03.31 ~ 04.04	Analysis (Theory)	OSP Stage 2030 – Analyze	#4
6	04.07 ~	Analysis	Flight Reservation System	#5

	04.11	(Case Study)	/ Library Management System	
7	04.14 ~ 04.18	Analysis (Discussion)	Team Project Presentation #2	
8	04.25		Mid-Term Exam.	
9	04.28 ~ 05.02	Design (Theory)	OSP Stage 2040 - Design	#6
10	05.06 ~ 05.09	Design (Case Study)	Flight Reservation System / Library Management System	#7
11	05.13 ~ 05.16	Design (Discussion)	Team Project Presentation #3	
12	05.19 ~ 05.23	Construct (Theory)	OSP Stage 2050 - Construct OSP Stage 2060 - Test	
13	05.26 ~ 05.30	Construct (Case Study)	Flight Reservation System / Library Management System	#8
14	06.02 ~ 06.05	Special Topic	Introduction to Formal Methods	
15	06.09 ~ 06.13	Construct (Discussion)	Team Project Presentation #4	
16	06.20		Final Exam.	