

**67-271 Fundamentals of System Development
Carnegie Mellon University
Syllabus
Fall 2002**

Professor: Cleotilde Gonzalez, coty@cmu.edu, Porter Hall 223C, 412-268-6242

Class secretary: Sharon Blazeovich, sharonb@andrew.cmu.edu, Porter Hall 223E, 412-268-3259

TAs: Smit Atal Gupta, sagupta@andrew.cmu.edu
Others: TBA

Credits: Undergrad (9 hours)

Lecture: Section A: TR 9:00-10:20 AM BH A51
Section B: TR 3:00-4:20 PM PH A18B

Office hours: Cleotilde Gonzalez, TR 1:00-2:00 PM and by appointment
TA office hours TBA

Overview

Fundamentals of System Development is an introductory course in software systems analysis and design and project management. It is a required course in the IS major and minor sequence. In this course you will learn the fundamental theory, methods and techniques needed to develop complex information systems projects.

This is not a programming course. Creating complex software products is a lot more than programming. It involves planning, gathering requirements, modeling business needs, creating blueprints for building the system, and managing and organizing resources. Software building is a challenging, difficult, complex and expensive activity. Because of its often intangible and intellectual nature, many authors in this field claim that developing software systems is the most complex engineering activity ever attempted.

In this course you will learn processes, methods, and techniques to plan, analyze, and design complex software products. This course is organized according to a Software Development Process (SDP) including phases common to many development strategies. SDP is used as a framework on which different pieces in the software development activities are glued together.

Software Analysis and Design are fields in constant evolution. Today, the most exciting ideas are in Object-Oriented (OO) techniques. The Unified Modeling Language (UML) provides a common set of diagramming techniques rich enough to understand different perspectives of an information system from the initiation to implementation. This course will present these OO ideas and techniques.

Fall, 2002

Objectives

Upon successful completion of the course, you should be able to:

1. Describe software development tasks and different approaches to software development
2. Determine technical, economical, and organizational factors to do a feasibility analysis
3. Apply information-gathering techniques towards the documentation of requirements for an information system solution
4. Understand how to use OO analysis and design methodologies and modeling techniques towards the documentation of a system solution with use cases and class models
5. Plan, design and execute usability evaluations
6. Develop project plans, and understand how to organize, direct, and control a software project

Textbooks and readings

Textbooks:

- [DWT] Dennis, Allan; Wixom, Barbara H.; Tegarden, David (2002). *Systems Analysis and Design: An Object-Oriented Approach with UML*. John Wiley & Sons, Inc.
- [SW] Schneider, Geri; Winters, Jason P. (2001). *Applying Use Cases: a Practical Guide*. Addison-Wesley.

Additional readings are on the schedule. The references for additional readings are:

- [RC] Rosson, Mary Beth; Carroll, John M. (2002). *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufmann.
- [BJR] Booch, Grady; Rumbaugh, James; Jacobson Ivar (1999). *The Unified Modeling Language User Guide*. Addison-Wesley.
- [BH] Beyer, Hugh; Holtzblatt, Karen. (1998). *Contextual Design*. Morgan Kaufmann.
- [NM] Nielsen, Jacob; Mack, Robert L. (1994). *Usability Inspection Methods*. Wiley & Sons.

We have prepared reading packets for each of you, containing the chapters required in the schedule. Packets will be available from Sharon Blazeovich in Porter Hall 223-E on September 10.

Organization

The course consists of lectures, readings, project assignments and exams.

Lectures are the main source of information that will help you meet the course objectives. Lectures cover general topics on Software Development as well as specific methodologies and tools such as feasibility analysis, information gathering techniques, OO analysis and design modeling, usability evaluations and project management. The class schedule shows the topics I intend to cover in each class.

Fall, 2002

Readings from the textbooks as well as reading packet are also presented in the schedule when they are due. The readings will help you expand and review information of the topics covered in the lectures. Please make sure to do your reading BEFORE coming to class.

Project Assignments will help you practice the concepts presented in the lectures and also help you prepare for the exams. You must work on projects individually and independently. You are not allowed to share any information concerning the solution of the project assignments with other students in this course. The project assignments consist of several deliverables due on the days presented in the schedule. More information on the project assignments appears after the schedule.

Exams. There will be 2 in-class exams and a final (see schedule; however, the date of the final is fixed by the Registrar). Exams will include multiple choice, short answer, short essay, modeling and diagramming questions. Exams will be based upon the readings, lectures, and assignments. I will hold in-class review sessions the class before the exams. Exams are to be taken individually. Mid-term exams are NOT cumulative, final exam IS cumulative.

Schedule

AUGUST	Lecture Topic	Reading Due	HW Assigned	HW Due
Tues, 8-27	Introduction to class, policies and project assignments.			
Thurs, 8-29	Fundamentals of Software Development: Software Development Process	DWT. Chapter 1 Pages 1-15		
SEPTEMBER				
Tues, 9-3	Project Initiation. Feasibility and Risk analysis	DWT. Chapter 2 SW. Chapter 1	Feasibility Analysis	
Thurs, 9-5	Requirements Gathering Techniques I: Interviews	DWT. Chapter 5 Pages 120-128		
Tues, 9-10	Requirements Gathering Techniques II: Questionnaires, Observation, Focus Groups, JAD	DWT. Chapter 5 Pages 128-147	Requirements Gathering	Feasibility Analysis
Thurs, 9-12	Requirements Gathering Techniques III: Contextual Inquiry	Reading Packet: BH. Chapters 3 and 4		
Tues, 9-17	Visual Modeling and introduction to UML	Reading Packet: BJR. Chapters 1 and 2		Requirements Gathering
Thurs, 9-19	TOC - NO CLASS			

Fall, 2002

Tues, 9-24	Principles of Object-Oriented Modeling	DWT. Chapter 1 Pages 15-24.		
Thurs, 9-26	Review for TEST 1			
OCTOBER				
Tues, 10-1	TEST 1 – Covers up to 9-24 class			
Thurs, 10-3	Use-Case Modeling I: Introduction and elements of Use Case Diagrams	SW. Chapter 2 DWT. Chapter 6 Pages 159-180	Use-Case Modeling	
Tues, 10-8	Use-Case Modeling II: Documenting Use Cases. Activity Diagrams	SW. Chapters 3, 4, and 5 DWT. Chapter 6 Pages 152-159		
Thurs, 10-10	Use-Case Modeling III: Reviewing Use Cases/Moving to Design	SW. Chapter 8		
Tues, 10-15	<u>Use-Case Modeling IV: Excercises</u>			
Thurs, 10-17	Class Modeling I: Find Classes from Use-Case Behavior	DWT. Chapter 7	Class Modeling	<u>Use-Case Modeling</u>
Tues, 10-22	Class Modeling II: Distribute Use-Case Behavior to Classes. Interaction Diagrams.	DWT. Chapter 8		
Thurs 10-24	Class Modeling III: Describe classes. Multiplicity	DWT. Chapter 7		
Tues, 10-29	Architectural Design Reviewing the Design Model	DWT. Chapter 9 SW. Chapter 9		Class Modeling
Thurs, 10-31	Scenario-Based User Interface Design and Prototyping	DWT. Chapter 11 Reading Packet: RC. Chapter 1 and 6	User Interface Design	
NOVEMBER				
Tues, 11-5	Review for TEST 2			
Thurs, 11-7	WWW navigation and interface components TEST 2 – Covers up to 10-29 class	DWT. Chapter 12		

Fall, 2002

Tues, 11-12	Usability Evaluation I: Introduction to usability evaluation methods (UEMs) and Usability Aspect Reports (UARs)	Reading Packet: NM. Chapter 1	User Interface Design	
Thurs, 11-14	Usability Evaluation II: Heuristic Evaluation	Reading Packet: NM Chapter 2	Heuristic Evaluation	
Tues, 11-19	Usability Evaluation III: Think Aloud	TBA		
Thurs, 11-21	Project Management I: Planning and Organizing	DWT. Chapter 3 Pages 56-75	Project Management	Heuristic Evaluation
Tues, 11-26	Project Management II: Directing and Controlling	DWT. Chapter 3 Pages 75-87		
Thurs, 11-28	THANKSGIVING BREAK- NO CLASS			
DECEMBER				
Tues, 12-3	Invited speaker	TBA	Retrospective Due the day of the FINAL EXAM	Project Management
Thurs, 12-5	Review for FINAL			

Communication

We will be using the University-supported system for course delivery called Blackboard. You will be able to access class slides and project assignments, read announcements, and hold discussions about the course, etc. through Blackboard. To get started, go to <http://www.cmu.edu/blackboard/help/help/index.html> and follow the instructions about logging in and accessing class information. Our class is found under F02-67-271 Fundamentals of System Development.

Please make a regular habit of checking the Blackboard for this class because all course announcements will appear there!!! It is your responsibility to be informed of whatever is posted there.

Project assignments

The purpose of the project assignments is to give you experience in the different systems development concepts, having to work on the analysis and design of the same product over the semester. Throughout the semester you will create a "Project binder," made of the project deliverables. A description of the case to be used throughout the course is available in Blackboard. Each project assignment will give you more information towards the development process. Also, each project assignments will give you practice in using modeling techniques, a required skill for any IS developer.

Project assignments should be done individually and independently. Cheating will not be excused and will lead to failure in the course. I expect quality reports. Points will be subtracted for unprofessional project assignments. An "A" project is complete, correct, and

Fall, 2002

convincing in every respect; it is a pleasure to read. It demonstrates the author's initiative as well as thoughtfulness, insight, and depth of analysis into the methodologies and techniques. It goes well beyond the minimum requirements of the assignment. It meets or exceeds my expectations in every dimension. An "A" project could be used, without modification, to demonstrate the concepts learned in the class to new students.

Project assignments must be delivered in your digital drop box in Blackboard. No paper should be handed in. Project assignments are due at 9:00 AM, the day appearing on the syllabus. That is, no assignments will be graded if not found in your Blackboard electronic mailbox by 9:00 o'clock. Therefore do not try to submit the project assignments at 8:59 AM. Make the necessary arrangements to make sure we will get your assignments by 9:00 AM, not 9:01, not 9:30, not midnight!. An assignment is classified as late if received after 9:00 AM on its due day. An assignment is classified as missing if not received within 24 hours of its due day.

A good management practice in IS is to keep track of the time it takes to perform each activity. A time sheet template will be available in Blackboard. You are required to submit your project assignments with a time sheet. A project assignment will not be considered for grading without the time sheet.

Grading

Point system. In order to make the grading as systematic and equitable as possible, I use a point system in which each project assignment and the exams are given a certain number of points. The maximum number of points is 105. Since you can get an A+ if you have 98 or more points, there are surplus points available in case you need to miss an assignment.

Therefore we do not excuse or discuss late or missing project assignments or missed examinations. Late project assignments will automatically reduce your grade by 20% of the points of the assignment. For example, you will loose 2 points if you are late in an assignment that is worth 10 points.

The only circumstances in which there may not be a penalty for late or missing project assignments or missed examinations is a documented medical emergency or death in the family. Job interviews and plant trips are not valid excuses for any reason.

	Points
Project assignments	
1. Feasibility Analysis	5
2. Requirements Gathering	10
3. Use-Case Modeling	10
4. Class Modeling	10
5. User Interface Design	5
6. Heuristic Evaluation	5
7. Project Management	5
8. Retrospective	5
Exams (individual)	
Test 1	15

Fall, 2002

Test 2	15
Final examination	20
Total	105

Course Policies

General policies for the IS program courses are in:

<http://www.andrew.cmu.edu/course/67-272/Behavior.html>

This site includes policies for:

Study Skills and Time Management
Personal Software Process Documentation
Regrades
Missed Examinations and In-Class Projects
Use of Email
Office Hours
Respectful Behavior
Academic Honesty and Integrity

Please make sure you read carefully and understand these policies. These policies will be explained to you in the first class session of the course 67-272. Additional policy items, particular to this course are stated next.

Regrading. I will accept regrade requests only if submitted within **one week** after the project assignments/exam has been handed back.

Attendance and class participation. I do not give grades for class participation or attendance. I simply expect your attendance and engagement in the class. Research shows the best predictor of grades in college is class attendance. You are responsible for all information presented and discussed in class.

Sections. In order to better understand your learning needs, I have separated this course into two sections, A and B. Room A18B does not fit more than 50 people. So, please DO NOT attend a lecture other than the section you have registered for.