

CSE 360
Introduction to Software Engineering
Spring 2012

SLN-12784

General

Building	BYAC
Classroom	150
Lecture days / hours	M, W / 3:30 – 4:45 pm

Textbook

Required:

Pressman, R.S., 2010, *Software Engineering: A Practitioner's Approach*, 7th Ed., ISBN: 978-0-07-337597-7, McGraw-Hill Higher Education.

References:

UML Standard, <http://www.uml.org/>

Instructor

Hessam S. Sarjoughian
Brickyard/476
URL: <http://sarjoughian.faculty.asu.edu>
Office hours: Tu, 2:45 – 3:45; W, 1:15 – 2:15 pm; and by appointment.

Teaching Assistant

Yuan Wang
Office hours: M, Th, F, 1 – 2:30 pm
Office location: Brickyard/, 425/426

Course Description (CSE/ASU catalogue): Software life cycle models; project management, team development environments and methodologies; software architectures; quality assurance and standards; legal, ethical issues. Prerequisites: CSE-210 and CSE-240.

Course Objectives: (I) Students will develop an understanding of software engineering topics including basic software engineering approaches supporting requirements, design, coding, testing, maintenance and quality assurance. (II) Students will become aware of national and international standards. Students employ a disciplined process for acquiring requirements, develop software design, reasoning about the correctness of software programs while adhering to a code of ethics. (III) Students will learn to work in a team setting where project planning and tracking is used as a basis to support systematic software development, software inspection, software product documentation and oral presentation.

Course Project: The purpose of the course project is for a team of students to individually and collaboratively apply the course concepts, techniques, and approaches project in creating a software product. Each team consists of 4 students. Each team must have a leader throughout the semester unless there is a justification for a change. Teams formed by students must be approved by the instructor. For details refer to the accompanying Project Booklet.

Software Tools and Computing Laboratory: Projects are to be developed using Rational Development Suite and a suitable IDE. Computers, software tools, and study rooms are available in Brickyard, room 214. For hours refer to the course webpage.

Course Objectives and Outcomes

1. To develop an understanding of software engineering topics.
 - 1.1. student will understand basic software engineering approaches for requirements, design, coding, testing, maintenance, and quality assurance.
 - 1.2. student will develop an awareness of national and international standards.
2. To develop software following a disciplined development process.
 - 2.1. student will be able to elicit and document problem requirements.
 - 2.2. student will be able to create an architecture to solve a problem.
 - 2.3. student will be able to reason about program correctness.
 - 2.4. student will be able to follow a code of ethics.
3. To work effectively on a software development team.
 - 3.1. student will plan and track a software development effort.
 - 3.2. student will be able to conduct a software inspection.
 - 3.3. student will be able to present software product and process results in oral and written form.

Curriculum (topics covered)

Students will work **individually** and in **groups** to learn software engineering concepts, theories, and practices.

Software life cycle models

- Commonly used process models
- Strengths/weaknesses of process models
- Awareness of international software development standards

Project management

- Approaches for increasing software quality and productivity
- Plan and track a small software development effort
- Apply software risk management techniques

Team development environments and methodologies

- Software requirements management and use-case generation
- Using Rational Rose for creating UML based models
- Integrate software that conforms to interface specifications in a team environment

Software architectures

- Practice OO software design from user requirements

Quality assurance, standards, legal, ethical issues

- Process measures (for quality and productivity)
- Participation in software inspections
- Software testing
- Software engineering code of ethics

Grading

	% of Course Grade	Date [§]
In-class exercises and participation	5%	During lecture
Homework	15%	Due at start of class
Midterm exam	20%	03/07 start of class
Final Exam	20%	Wed., 05/02, 12:10 – 2 pm
Project	40%	See Project Booklet
Total	100%	

§ Dates are subject to change; final exam date and time are fixed.

Holidays

Martin Luther King Day	Jan. 16
Spring Break	March 19 - 25
Last day of classes	April 24

Withdrawals

Note that students wishing to drop the course AT ANY TIME must take appropriate actions according to University policy and guidelines. **Ceasing attendance does not automatically drop you from the course.**

University 21st Day	Jan. 25
Course Withdrawal Deadline - In Person & Online	March 28
Academic Status Report #1 (tentative)	Feb. 1 - 8
Academic Status Report #2 (tentative)	March 19 - 25

Project

Tasks / Activities	% of Grade	Date/Time
Presentation 1		02/27 and 02/29
Preliminary report + prototype	20%	03/02, 3:30 pm, CIDSE Front Desk, Blackboard
Presentation 2		04/16 and 04/18
Final report + video + software	20%	04/23, 3:30 pm, in-class, Blackboard
Total	40%	NA

Attendance: Participation is an integral part of the course and attendance will be monitored randomly. Every student is responsible for all announcements made during class and posted on the course webpage.

Homework: Each homework assignment must be turned at the start of the class on its due date. **There is 20% reduction per day for late submissions.** Select problems from each homework assignment will be graded. The lowest homework assignment and in-class exercise grades are excluded from the course grade.

Project: Evaluation of all deliverables – reports, presentations, and software – is based on their completeness and quality. **There is 20% reduction per day for late submissions.** The deliverables need to be typed. Peer evaluation will be used in grading projects. Each project report must include a table listing contribution percentages for all team members. Presentations and deliverables must be submitted in hardcopy and their softcopies uploaded to Blackboard. Proposal for extra credit up to 10% of the total project grade may be included. Proposal submitted as part of the Preliminary Report is acceptable.

Work assignment preparation: Software tools including word processing and Rational Rose UML tool are required to be used. Only typed project reports, presentations, and assignments are acceptable.

Grading: Course grade is based on the table below. **Students are responsible for all materials covered and discussed in class, posted on Blackboard, and email correspondences.** Examinations **may not be taken individually** except in special situations with prior arrangement and approval at least 3 working days (72 hours) in advance of the exam date. Presentations must be given on their assigned dates. **After the return of any graded course materials, there is a one-week period to request for reevaluation.** Each inquiry must state in writing the problems and the sources of errors. Original work must be submitted with the inquiry in hardcopy at the beginning of class or emailed to the instructor and TA.

% total score	≥97	≥90	≥87	≥80	≥77	≥70	≥60	<60
Letter grade	A+	A	B+	B	C+	C	D	E
Points for GPA	4.33	4.00	3.33	3.00	2.33	2.00	1.00	0.00

Academic Integrity and Ethics: All instances of alleged cheating will be handled by the Dean's office according to the [Student Code of Conduct and Student Disciplinary Procedures](#) and [STA 104-01](#). This university policy will be applied to all work submitted for grade, including homework assignments, quizzes, exams, reports, presentations, and software development. **Note:** you are encouraged to discuss class assignments and other activities with your instructor, teaching assistant, and fellow students. However, all work submitted as part of course work must be your own. I.e., final work submitted by student must represent his/her own individual or team efforts unless stated otherwise by the instructor.