**CPE 402 Software Requirements Engineering**
**Fall 2012**


**Instructor** David Janzen
**Website** http://www.csc.calpoly.edu/~djanzen/courses/402F12/
**Email** djanzen@calpoly.edu
**Office** 14-212
**Phone** (805)756-2929
**Office Hours** Monday, Thursday: 3:10-4:30pm; Wednesday: 10:30am-noon; Friday: 4-4:30pm


## Course Description

This is the first in a three-course (academic year long) sequence. It is generally assumed that the student plans to take all three courses in the sequence in one year. The primary objective of the year-long sequence is to develop a software system for an outside customer. The primary objective of this course is to work with a customer in teams to develop a Software Requirements Specification and High-Level Architecture to serve as input to CPE 405 Software Construction.

The catalog describes this course as follows:

> *Software requirements elicitation, analysis and documentation. Team process infrastructure and resource estimation to support appropriate levels of quality. Software architectural design.*

## Objectives

- To learn skills required to produce and maintain a high-quality software product on time and within budget
- To know and execute principles and concepts of software requirements engineering
- To know and execute principles and concepts of software architecture
- To work effectively as a member of a team to meet project milestones
- To understand and apply a software process
- To effectively write and speak about software engineering

## Prerequisites

CSC/CPE 305 and either CSC/CPE 307 or CSC/CPE 309; or permission of instructor.

## Required Texts

1. Wiegers, Software Requirements, 2nd Edition, Microsoft press, 2003
2. Fowler, UML Distilled, 3nd Edition, Addison-Wesley, 2003

## Recommended

Gold Fibre Design: Project Planner Notebook No. 20-817 to be used for your status reports

## Additional Reading

Periodically, additional articles will be passed out or assigned for you to find and read.

**Schedule**
This course will meet Monday, Wednesday, and Friday from 1:10pm to 3pm in 14-232A. Students are expected to attend all course meetings.

Typically the first hour will include instructor and student presentations and discussions. The second hour will generally be spent on project activities, some prescribed and some delegated to team discretion. Fridays will generally include a remote or face-to-face meeting with the project customer.

A tentative schedule of topics and activities will be available on the course website. This schedule is subject and likely to change. All reading assignments should be completed prior to class as noted in the schedule.

**Communication**
The best place to discuss the course is during lecture and laboratory times. The main communication tools for the class will be the course schedule, a Piazza discussion group, and PolyLearn. Students will be expected to check the first two on a daily basis. Posts to Piazza should never criticize people. Constructive criticism of artifacts and ideas is acceptable. All assignments will be placed on the course web site and/or announced in lecture. Most class materials are available on the course web site; be sure to check regularly. Grades will be posted in PolyLearn.

Email will only be used for special circumstances, such as communicating time sensitive information or personal issues. All students are expected to have their calpoly.edu email accounts forward to wherever they will read email at least daily. If you use email, put CSC 402 on the subject line to get the best response time. Leaving voicemails should be a last resort.

**Classroom Etiquette**
To ensure a professional learning environment, the following rules will be enforced in the classroom:

- Do not eat except when food is provided for the entire class
- Do not use electronic devices that make sounds (e.g. cell phones, ipods)
- Do not use computers for anything besides presenting or taking notes when anyone is presenting

**Grading**
The course grade will be determined on the following factors:

1. Examinations (20%)
2. Attendance and Participation (10%)
3. Individual Assignments (25%)
4. Team Project Artifacts and Presentations (45%)

Letter grades will be assigned based on the expectation that an ‘A’ is earned with excellent work on all aspects of the course, sustained throughout the course. A ‘B’ is earned with very good work, perhaps excellent at times. A ‘C’ is earned with average work, perhaps very good at times but poor at other times. A ‘D’ is earned with consistently poor work.

**Examinations**
Written midterm and final exams will be given. Each exam is worth 10% of the final grade. The midterm exam will be given on October 24 from 2pm to 3pm. The final exam will be given on December 3 from 1:10pm to 4pm. No late or makeup exams will be given.

**Classroom Attendance and Participation**
Students are expected to take an active role in their own learning and the learning of their peers. Students will receive 1% credit for each week of acceptable participation. Acceptable participation is earned by attending, arriving on-time, participating in discussions and activities, and being prepared for all lecture and lab sessions, including reading all assigned sections prior to class. Students should make it a habit of adding meaningfully to discussions and asking relevant questions without dominating discussions. Up to three absences will be acceptable provided the student provides a valid reason, notifies the instructor by email, and receives an acknowledgement prior to class. More than three absences, for whatever reason, will result in a zero for the course participation grade.

**Individual Assignments**
Quizzes (5 x 1% = 5%)
Iteration Time Reports (5 iterations x 1% = 5%)
Weekly Status Updates (9 x 0.5% = 4.5%)
Presentation (2%)
Resume/Cover Page (1%) Due 9/21/12
Requirement Sample (2%) Due 10/10/12
UML Diagrams (2%) Due 11/09/12
Critical Analysis (2%) Due 11/16/12
Self/Peer Evaluations (1.5%) Due 12/03/12

**Iteration Task/Time Reports**
Students must track the tasks and time they spend on the group project. Students must enter all tasks, time estimates, and actual times for each iteration. At the end of each iteration, a detailed report should be saved in pdf format and emailed to Dr. Janzen by 10am on the Wednesday following completion of the iteration. Task/time entries will be graded pass/fail.

**Weekly Status Updates**
Each student must submit a brief weekly status update to Dr. Janzen. The update is an opportunity to share individual accomplishments and challenges on the course project. The update will be submitted through a web form and is due by 5pm every Friday. The updates will be graded pass/fail.

**Individual Presentations**
Students must give one short (5 to 7 minute) presentation in class. A list of suggested topics and dates are provided on the course schedule. Students must select a topic/date by September 24. Students are to speak without notes, although strategic visual aids may be used. Presentations will be graded subjectively both on the quality of the content and the effectiveness of the communication.

**Resume/Cover Page**
Develop a resume and cover letter as though you were applying for a position in a company. In the cover letter include what team aspects most interest you (e.g. Team Lead, Requirements, Quality Assurance, Architecture, Coding, Metrics). Be sure to highlight related experience. Sell yourself! You may also specify other students whom you do or do not want to have on the same team. These requests may not all be honored, but will be taken into consideration when assigning teams.

**Requirements Sample**
Students must turn in one persona, one use case, one functional requirement specification, and one non-functional requirement specification. Each of these should be created for the team project, but must be created individually. Coordinate with your teammates to avoid duplication.

**UML Diagrams**
Students must turn in two UML diagrams. The diagrams must be two different types of diagrams (e.g. class and sequence). Each of these should be created for the team project, but must be created individually.

**Critical Analysis**
Students must work with an assigned partner to document a critical analysis of an alternative design. The analysis should generally address questions such as:

- Are there technical problems with this design?
- Are there major omissions in this design (e.g. functionality, security, performance)?
- Is it feasible to implement this design by this team in the timeframe allotted?
- Does the design satisfy the customer's functional requirements?
- Does the design satisfy the customer's non-function requirements/quality attributes?

Hand in two printed copies of your analysis on the due date. The analysis should be a numbered list of items. Each item should note the source for the comment (e.g. Design Spec section 3.2.1), along with the criticism and suggestions for possible improvements. Group your review items into the following categories: Technical Risks, Information Risks, Economic Risks, Managerial Risks

**Self/Peer Evaluation**
Each student will complete a self/peer evaluation. As part of the final evaluation, students should detail their individual contributions to the project and identify any particular roles the individual fulfilled (e.g. recorder, team lead, quality assurance), and when they fulfilled that role (e.g. all quarter, weeks 3-5). They should identify all significant contributions to artifacts. Identify what role you took on the artifact (e.g. author, editor, reviewer). Finally, any significant tasks completed on the course project should be identified (e.g. researched Hibernate and made presentation to team).

**Team Project**
All of the work in CSC402 is connected to a three term (one year) project. You will work in a group to carry out each phase of the project. The project itself will have to meet standards of the instructor and the customer. The project is the cornerstone of this course and is the largest basis of your course grade. Teams of approximately four to six members will be formed early in the course. Team members will be internally managed to deliver a number of project artifacts. Most significantly the teams will produce a Software Requirements Specification, a High-Level Architecture, and a system prototype. The team will present the requirements, architecture, and prototype to the class and the customer. Project artifacts will be **due at noon one day before** the iteration end date specified on the course schedule.

The project evaluation is done as if you were a corporate employee. The project grade is assigned subjectively on an individual basis. Criteria used in determining the project grade will include action item acceptance and timely completion, quality of artifacts, self and peer evaluations, perceived leadership and teamwork skills, and quality of presentations. Individual artifacts will typically not be assigned a grade. ***Coat-tail hanging or non-performance by an individual will result in a course grade of F. You are required to participate fully in your group project.*** You must perform as part of a team. This is paramount. If you are not sure how you are doing on your team project grade, make an appointment to discuss it with Dr. Janzen.

Each project deliverable must be completed in a professional way. Work may be returned until it reaches a professional standard. You cannot receive a passing course grade until your portion of the project is completed satisfactorily. If substandard work is turned in toward the end of the quarter, all group members will earn a course grade of F.

**Team Project Tools**
Each team will be provided accounts with an appropriate set of project management tools. Each team must use these tools for planning and tracking project status, and keep this information up-to-date on a weekly basis. The team must publish the current version of all significant project artifacts including, at a minimum:

- Your team name, team logo, and team member names and contact information
- Lists of assigned action items for the entire team
- Source and test code
- Configuration and build files
- Project artifacts such as Vision and Scope, Iteration Task/Time Reports, Metrics, Software Requirements Specification, and High-Level Architecture

**Late Work Policies**
A software engineer has a responsibility to manage time effectively and turn in work on time. If you are having a problem, discuss it as early as possible before the due date.

For CSC 402, the following nominal late policies apply:

- Individual Assignments
  Homework assignments are due at the beginning of class on the date specified. Late status updates or homework assignments will be given a maximum of half credit and accepted only until the next class period following their due date.
- Project deadlines
  Project deliverables must be turned in on time, even if incomplete. If a significant part of a deliverable is missing or unacceptable, the individual(s) responsible will be penalized 3 grades (e.g. A to B). Unacceptable deliverables may be resubmitted, without penalty, up to one week later. Failure to resubmit work or subsequent resubmittals will be penalized one grade per week or partial week.
  Note: depending on circumstances, project deliverable penalties are assessed on an individual or group basis. If your group has a non-performer, turn in your deliverables on-time with an accurate credits section. The identity of the non-performer will be crystal clear.

**Integrity**
All work submitted is to be your own. Cooperative study and mutual aid are healthy learning methods and are strongly encouraged. You are especially encouraged to work with other groups. Just cite sources of anything you have copied, summarized or discussed directly with another. It is cheating to copy someone's work or allow someone to copy your work. It is cheating to copy material from a publication without giving credit. Plagiarism will result in a course grade of F. When you find good ideas by other people, the best policy is to summarize other work in your own words and cite their work as the source for the principle you state. Citing resources is not a sign of weakness of your own ideas, it is a sign that you can do research and build on others' work.