COMP 586 Object-Oriented Software Development (OOSD). Fall 2010

Dr. Shan Barkataki, Computer Science Department, CSUN
email: shan @ csun . edu **(preferred) Office : JD 4449 (818-677-2733)**

COMP 586	17752	M	19:00 – 19:45	JD3508

OFFICE HOURS: M 15:00-16:15 W 15:00-16:15, email at any time, I will generally respond within 24 hours (excluding weekends and holidays).

COURSE SYNOPSIS: In this course you will learn about analysis and design modeling using objects oriented methods. The modeling language (not programming language) will be UML-II, which is the industry standard modeling language for expressing software design. You will study the various static and dynamic modeling elements available in UML-II. The course will also cover some fundamental design patterns and discuss model based software development. You will participate in a group project to create analysis and design models for a moderately complex software intensive system. You will also write a research paper on a current topic in object-oriented software development. Expect to spend a minimum of 6 hours per week for this class (3 hours class time, 3 hours doing homework, research, project work and self-study). This course will make use of Moodle. The text book will be used along with course handouts and some online assets. In this syllabus the term coursework refers to all homework assignments, project assignments, research papers, quizzes and exams.

PREREQUISITES

1. Comp380/L and Comp322/L or equivalent. Concurrent enrollment in the prerequisite classes is not acceptable. Proficiency in programming skills.
2. Proficiency in reading and writing in English.
3. Basic understanding of the object-oriented programming principles. Recommended but not essential: CS 432 - Object Oriented Programming.

TEXT BOOK AND COURSE NOTES: Applying UML and Patterns by Craig Larman, 3rd edition Prentice Hall 2004. ISBN 0131489062. Course handouts in Moodle.

MOODLE: You will be auto enrolled into Moodle by the CSUN IT department. ***I would like to request you to post a passport type photo into your Moodle profile so that we all get to know you.*** Course notes, discussion sessions, papers, and all coursework will be posted in moodle.csun.edu. Moodle will be the definitive source for all coursework assignments and updates, class announcements, schedule changes, and syllabus updates. Typically you will also get an email notification. In case of any conflict arising between what is stated in Moodle and other sources, contents in Moodle will prevail.

CSUN EMAIL: I will post class related announcements in Moodle; these will also arrive as emails in your CSUN email inbox. If you do not use the CSUN Webmail client, then please set up webmail auto-forwarding. It is your responsibility to ensure that you receive and read the emails that arrive at your CSUN email address. In case of email problems please contact the CSUN IT help desk. Also make sure that your email accounts are not over quota due to junk mail etc.

TURNITIN.COM: Some coursework will have to be submitted at www.turnitin.com. You will need to self register at this site. I will provide the class code and the enrollment password in class.

Instructions & tutorial at http://www.turnitin.com/
When you register at turnitin.com, please use your CSUN email address, and your first name & last name as they appear in Moodle. If your turnitin.com name & email do not match those in Moodle then your grade may not import correctly into Moodle resulting in a lower class grade.

CLASS ETIQUETTE: *Please silence all pagers and cell phones before entering the classroom.* When in class, **PARTICIPATE**. No snoozing in class please; it is not professional behavior and shows disrespect towards me and your colleagues. If you feel sleepy, then take a walk outside and return when you recover. OK for you to stand at the back of the room to beat that drowsy feeling!! Read "How to ace my class" in my website.

COMPUTING DEVICES DURING QUIZZES & EXAMS: Use of **c**omputing devices is not permitted during quizzes and exams. All computing devices (including cell phones & PDAs) must remain off, or in standby mode, throughout the duration of the quizzes and exams.

EVALUATION CRITERIA

Each coursework will earn points in the range -33 to +100. How does a coursework earn negative points? See items 1 and 2 under **GRADING POLICY**. The Semester Total Class Points is a weighted sum of the individual coursework points, computed using the formula shown below.

Semester Total Class Points = ΣCoursework_point(i)*Coursework_weight(i)

The coursework weights are shown as percentages in the table below.

Two Quizzes	Weights: first 15%, second 20%. May include material from the project & other assignments. Quiz 1: Sept 20th Monday, Quiz 2: Nov 1st Monday
One Final	Weight: 25%. Final, may include material from the quizzes, project and other assignments. Monday, Dec 13, 2010 08:00 – 10:00 PM
Project: Individual.	There will be multiple assignments. Total weight of all project related individual assignments: 10%. See Project Guidelines in Moodle.
Project: Group	There will be multiple assignments. Total weight of all project related group assignments: 10%. See Project Guidelines in Moodle. Actual score earned by a student = (Group score * Individual score)/100
Research paper	Weight 10%. See Research Paper Guidelines in Moodle
Homework	There will be multiple assignments. Total weight of all individual homework assignments: 10%. Some project assignments may serve as homework.
Attendance and participation	Regular and punctual attendance is essential for passing this class. Coursework may include material discussed in class meetings that is not in the textbook or the handouts. Please arrange for a "buddy" to brief you in case you have to miss class on odd occasions.
Makeup	Makeups are not given.

CHANGES TO THE SYLLABUS: This syllabus may be changed, with notification.

GRADING POLICY

1) You will earn most points by submitting coursework on time. You will lose points by submitting coursework late, at a rate of 25% per day unless you have made ***prior arrangements*** with me by email; verbal agreements do not count. You can request a reasonable extension, for good cause. Except in case of emergencies, a request for an extension needs to be submitted by email ***before the due date***. The extension request should specify: your name, description of the coursework, why you need an extension (in two lines), and the new date by when you will submit the work. Coursework submitted within the extension period will not incur any late penalty; however you will earn no points from a Coursework submitted after the solution has been discussed in class.

2) You are required to participate in and submit all coursework. Each missing coursework will attract negative points equal to 33% of the maximum. Negative points will reduce your Semester Total Class Points quite significantly, see Evaluation Criteria above. Note that even a poor solution will typically earn some points; in the worst case a zero; whereas a missing solution will earn -33 points.

3) You will earn more points by submitting professional quality work. Computer printout from CASE tools is best for model prints (printers available in all computer science labs). You will lose points by submitting work with spelling or grammatical errors, scruffy writing and drawings, work that is hastily put together without much thought, analysis, or care.

4) Please arrange to take all quizzes and exams on the scheduled dates. I will reschedule quizzes and exams only for certified emergencies, not for convenience.

5) In fairness to all, I don't give make-up work to compensate for poor performance in scheduled coursework.

6) Section entitled "ACADEMIC DISHONESTY" in Appendix C of the CSUN catalog gives definition of plagiarism and cheating; this entire section is included in this syllabus by reference. In this class, the term plagiarism also includes the case where a student submits any material for grading that is authored by someone else as his or her own work. Examples of such plagiarized work include term papers, computer programs, design solutions, and answers to questions written by a commercial entity, as well as the student's friend, relative, or tutor. There will be ZERO TOLERANCE against plagiarism and cheating. Students who cheat or plagiarize will earn the grade F in the coursework; there will be no opportunity to resubmit the plagiarized coursework or retake the quiz or exam.

7) Some coursework can be done in collaboration with another student.

   a) If you are going to do any coursework as a pair, then please email me in ADVANCE at shan@csun.edu requesting permission. Give your name, your partner's name, and the name of the coursework.

   b) There should be only one formal submission of the work and the names of both students should appear in the submission.

   c) When submitting at turnitin.com, one student should submit the real work the other student should make a dummy submission with a note giving the names of both partners and the title of the coursework.

d) Both partners must participate and contribute equally to the solution and each must have a copy of the solution.

8) I may call upon any student to explain any aspect of a coursework handed in for grading, including those done in collaboration with another. If you cannot explain the solution then you will earn a lower grade, including the grade F.

9) Submitting work done by someone else is cheating and will be considered as academic dishonesty. In all such cases a grade of F will be assigned.

10) In developing solutions you will earn more points by developing simple solutions and by using a disciplined approach.

11) All analysis and design work products should be expressed in standard UML-2- no inventing your own language!

SUBMITTING WORK

1. Coursework designated for submission at turnitin.com cannot be submitted by email or as hardcopy.
2. When a hardcopy submission is required, please submit it in class, not in my office or in the comp science office. Each hardcopy submission must have your last name and first name, printed on the top right hand corner on the first page. This also applies to work done in a pair.
3. Please do not send your submissions as attachments to emails, I have a very aggressive mail filter that will most likely send it to the bit bucket. Emailed submissions will not serve as proof of timely submission.
4. I expect all coursework (except quizzes and exams) submitted for grading to be printed and of professional quality. Scruffy drawings and hand written answers are not acceptable. See **GRADING POLICY** above**.**

TOPICS

- Overview of object-oriented development
  - Review of software modeling concepts
  - Development methods, processes, and life cycles
  - Introduction to UML
    - Small things- Classes, objects, interfaces
    - Relationships- association, aggregation, Inheritance and polymorphism.
  - Functional vs. object-oriented modeling
  - Agile software development process
- Analysis Modeling
  - Event-Response modeling
  - Use case modeling
  - Activity Diagrams, swim lanes
  - Requirements specification – style, clarity, format, tools (self study & research)
  - Case studies from text
- Static Modeling
  - Class diagrams showing relationships,
  - Interfaces & ports
  - Big things- Packages, components and subsystems
  - Demeter's principle

- Deployment diagrams
- Dynamic Modeling
  - Modeling object behavior
  - State machines & state charts
  - Modeling complex behavior with state charts
  - Nested states and state inheritance
  - Modeling internal and external concurrency
  - Case study from text- heart pace maker (no course notes)
- Modeling Interactions
  - Advanced Interaction and communication diagrams in UML-II
  - Loops and branches
  - Lifeline & interaction decomposition- fragments & references
- Modeling for real-time behavior
  - Timing diagrams
  - Advanced activity diagrams
- Architectural Design
  - Logical vs. physical modeling
  - System development process
  - Domain modeling
  - Modeling whole systems
  - Concurrency considerations- UML-II Task diagrams
  - Real-world concurrency- Operating system, threads, processes, IPC etc.
- Detailed Design
  - Parametric classes
  - Frameworks & design patterns
  - Common design patterns
    - GOF patterns
    - Describing design patterns in UML-II
    - Using design patterns in UML-II
- Other issues
  - Model driven Architecture
    - CASE tool issues