CS3337 Syllabus

Instructor: Dr. Jose M. Macias (Call me Jose)
email: josemasia@gmail.com
Phone #: cell (626)429-1714; work (818)393-0771
Jose’s Hours: By appt only. Mo-Th: 5-6pm. Fri: 9am – 8pm. Sat 9-11am.

Alternate Instructor: Richard Cross
email: richard.csdept@gmail.com
Richard’s Hours: Prefer appt. Mo: 3-5pm; W:1-3pm; Sat: 9-11am

EMAIL HEADING: Your email subject line SHALL include “cs3337 - (Meeting days) MW or SAT - Subject”

Office hours: Make an appointment by email/phone (Jose, or Richard).

Course number CS 3337
Course name Software Design
Credits 3 units
Contact hours 5 hours/week
Coordinator Prof. Raj Pamula

Pre-requisite: Strong programming skills using Java, or C++, or C or ?.

Text books
ISBN-10: 1284106004
References: “Software Engineering”, Ian Sommerville, Addison Wesley

Course Information Catalog Description: Methodologies and tools for requirements analysis and design of large complex software system, process models, project planning, tracking, documentation, communication, and quality assurance; group laboratory project; oral and written presentations. Lecture 2 hours, laboratory 3 hours. Graded ABC/NC.

Prerequisites: CS 2013 “Programming with Data Structures”.

Required/Elective: This course is required in the BS program.

Course Goals The Student Learning Outcomes (SLO) addressed by the course are:
SLO #5. Students will have the training to analyze problems and identify and define the computing requirements appropriate to their solutions.

SLO #6. Students will have the training to design, implement, and evaluate large software systems working both individually and collaboratively.

SLO #7. Students will be able to communicate effectively orally and in writing.

Other outcomes of instruction:
At the end of the course, students are able to

Estimate the cost and effort for software projects
Make schedules for software projects.

Elicit software requirements
Create data model, flow-oriented model and behavior model

Convert the requirement models into software architectures
Implement component-level design

Brief list of topics to be covered

Estimation for Software Project
Software Process
Requirement Engineering
Elements of Design Engineering
Project Implementation
Project Presentation and Defense
Project Version Control using Git and GitHub

Laboratory Projects
The students will be randomly divided into groups of 4 students per team. The students will complete a project based on a selected topic from their survey or provided by the instructors. One group member uploads the project under “Software Design and Implementation Document” assignment on CSNS. Projects will include steps such as:

Build data model, flow-oriented model
Behavior model for the selected project
Design software architecture based on the requirements
Create the component-level design
Implement system components
Integrate system components
System Demo and Presentation

Format of the Software Requirements Document/Software Design Document (SRD/SDD)
The instructors will provide the Department’s approved Requirements and Design Documents Templates for the documents that will be created during the semester.
To receive a grade in this class each part must receive at least a passing grade before the grades are weighted, that is, Exams, Quizzes, Readings, Documentation and Peers all must pass independently BEFORE they can be weighted. See below the grading curve.

**Exams**

There will be two pre-prepared exams. The exams are open book/notes. The exams are weighted 15% (Midterm), and 30% (Final).

**Quizzes on Readings**

Readings are a most fundamental part of the class. Readings are controlled by means of quizzes (6-8 pre-prepared questions in 20 minutes). The minimum expected number of readings is 6. All reading are required. The average of all quizzes is 20% of the total grade. Readings may include:

- At least 3 additional selected readings.

**Project Grading Policy**

The “Software Requirements Document, SRD” and the “Software Design Document, SDD”, will be evaluated at 15%.

The last important grade is the peers evaluation grade at 20%. This grade is the average of the grade each member of the team assigns to the other team members. This grade is sent in a private email to the instructor. It reflects the opinion of the team members.

**Mandatory attendance**

A large portion of this class revolves around in-class project development. Therefore, this class requires 100% attendance. In addition, the student must be on time after each break and at the beginning of class. Three “late” account for a full day of missed classes. Absences more than a full week of classes will receive a final grade "NC".

**Academic Integrity**

Cheating will not be tolerated. Cheating on any assignment or exam will be taken seriously. All parties involved will receive a grade of “NC” for the course and be reported to the Department Chair for further action.

**Grading Curve**

- **A** : 90% or better
- **A-** : 80% or better
- **B+** : 70% or better
- **B** : 60% or better
- **B-** : 50% or better
- **NC** : below 45%