CS5337 Syllabus

Instructor: Dr. Jose M. Macias (Call me Jose)
email: josemasia@gmail.com
Phone #: cell (626)429-1714; work (818)393-0771

Alternate Instructor: Richard Cross
email: richard.csdept@gmail.com

TA: Yin Yin Chen
email: espitaj@gmail.com

EMAIL HEADING: Your email subject line SHALL include “cs5337”

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

Course number: CS 5337
Course name: Software Design
Credits: 3 units
Contact hours: 4 hours/week
Coordinator: Prof. Raj Pamula

Text books:
Reference Textbooks (not mandatory):
“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.

Course Goals: The Student Learning Outcomes (SLO) that are 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**

1. Estimation for Software Project
2. Software Process
3. Requirement Engineering
4. Elements of Design Engineering
5. Project Implementation
6. Project Presentation and Defense
7. 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 provided by the instructors. Project code will be uploaded and maintained on GitHub.

**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.

**Exams/Quizzes**

There will be three exams. The exams are open book/notes. The exams are weighted 10%, 10%, and 20%.

**Readings (classical papers on Software Engineering)**

Readings are a most fundamental part of the class. Readings are controlled by means of quizzes. The minimum expected number of readings is 6. All readings are **required**. The average of all quizzes is 20% of the total grade.

Readings may include:

4. Other selected readings.

**Project Grading Policy**

Project work will be evaluated at 20%.

The last important grade is the peers evaluation grade at 20%. This grade is the average of the grades each member of the team assigns to the other team members. This grade is sent in a private email to the instructors. It
reflects the opinion of the team members. This grade has a “veto” power over the overall final grade of the student.

**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 class. Absences more than a full week of classes will receive a final grade of F.

**Academic Integrity (University Policy)**  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.