COMPUTER ENGINEERING DEPARTMENT

Course: CMPE 226
Course Title: Database Design
Semester: Fall 2003

Instructor Information and Course Description

Instructor: Dr. M.E. Fayad, Office Engr 283I, m.fayad@sjsu.edu, (408) 924-7364
Web page: http://www.engr.sjsu.edu/~fayad

Course Meeting Place/Time: Lecture: IS 216, Tuesday 6:15 p.m. to 9:00 p.m. (18:15 to 21:00)

Office Hours:	**Monday:**	**11:30 a.m. – 1:30 p.m.**
	Tuesday:	**3:00 p.m. – 5:00 p.m.**
Other times:	**Send an e-mail to schedule an appointment.**	

Course URL: http://www.engr.sjsu.edu/~fayad /current.courses/cmpe226-fall03

Course Catalog Description

For over 30 years, computerized database systems have been developed and used to help computers manage the increasing amount of data that we store and manipulate. This course presents an overview modern database systems and some advanced issues.

Prerequisites: CmpE126/CS046B (?), (Data Structures & Algorithms), or instructor's permission -- Good background in the practical use of an object-oriented programming language is a plus.

Required Textbooks: 5-7 chapters will be covered from 1st book.

1. P. Revesz, Introduction to Constraint Databases , Springer, New York, 2002, ISBN 0-387-98729-0
2. C.J. Date and C.J. Date. An Introduction to Database Systems, 8th Edition, Pearson Addison Wesley, July 2003, ISBN 0321197844
3. Ramez Elmasri and Shamkant Navathe. Fundamentals of Database Systems, 4th Edition, Addison-Wesley, July 2003, ISBN 0321122267

Required Articles, Columns, Case Studies, and Patterns will be posted on the web later.

Supporting Textbooks:
Paul DuBois. MySQL Cookbook, 1st Edition, O'Reilly & Associates, November 2002, ISBN 0596001452

Other Resources: Instructor notes will be available on the course web page.

Course Objectives:

This course develops your understanding of database system **CONCEPTS** that are independent of any specific database system. As an analogy, if you took a driver's education class, you learned there driving, which is independent of any specific car. This distinction between concepts and tools is important to keep in mind in this course (especially by students who like to list on their resumes a long list of specific operating systems, programming languages, and database systems).

These concepts include: data abstraction levels; data independence; data models, including relational, constraint, spatial, and spatiotemporal data models; query languages, including relational algebra, and SQL; database design concepts, including integrity constraints; and interoperability, including data and query interoperability. Many of these concepts have a remarkable staying power, while the specific systems change all the time. In fact, the more basic the concept the more likely it is to endure. For example, the concept of data abstraction endured even as new data models were invented: hierarchical in the 1950s, network in the 1960s, relational in 1970, and various spatial data models used in GI in more recent years. As another example, the QL language, invented in the early 1970s,is still the primary language of relational database systems today, whether they come from Oracle, IBM, Microsoft, MySQL, ybase, Empress, etc.

By understanding well the concepts and not just a specific system on a specific platform, you should be able to adjust better to future changes in the database systems area. In particular, GI is emerging as a very important extension of relational databases, with some estimates putting the word-wide GIS market at about one billion dollars annually, while the relational database market is about six billion dollars annually. Hence we will learn about GIS too in this course. This course will contain three projects that are designed to illustrate a variety of uses of database systems (**all subject to change**):

Project 1: This project will focus on relational databases and QL. For this you can use either the MLPQ system, which was developed at UNL, or the My QL system, which is a free relational database software. However, the classroom examples will be based on MLPQ.

Project 2: This project will focus on constraint and spatial data storage. For this you need to use the complete MLPQ/PRe TO system, which is like a GI system, and students in the past used it for various interesting projects that needed to store maps or drawings. Note that for this you cannot use MySQL.

Project 3: This project will focus on web accessible MLPQ databases. You need to set up a webpage with embedded QL queries that can remote access an MLPQ database.

BACKGROUND: This course is targeted to those individuals who are interested in database design and would like to learn several database design and systems.

Course Outline

Check Course Outline.

Grading Policy

Your grade in this course will be based on your performance on written homework, test, and a programming project.

4 Assignments	60
Midterm	45
Final Exam	60
2 Team Projects	60
Total	225

Your lowest homework grade will be dropped hence the maximum will be 210 points that you can earn without extra credits. Your final grade will be based on the percentage of your total points (200 points counts as 100%). **Your grade will be based on an overall curve.**

Final Grades:
Letter grades will be assigned at the end of the course. Final grades will be based on a competitive curve. Graduate and undergraduate students are graded separately. Students will be informed of their standing at intervals throughout the course. Final grades *are not* ***negotiable.*** Unless there are mathematical errors, I will be unavailable to discuss final grades. Borderline cases will be considered with extreme care, and fair grades will be rendered.

Class Attendance:
Class attendance is mandatory. If you have more than four unexcused absences, then you will be dropped from the class.

Due Dates:
Late homework assignments, extra assignments, and projects are NOT ACCEPTABLE. **There will be no make up tests.**

Group Projects:
The class will be divided into groups of 2-3 (three preferred) for team projects. Students will be responsible for forming groups. Students will give a final presentation of their project work. Grading criteria and project ideas will be posted in a project Web page.

On occasion, students take advantage of group work, letting other members perform the bulk of the work while they reap the benefits of a good grade and can spend more time on other classes. This happens only occasionally, but it will not be tolerated in this course. Two policies will help prevent this:

1. **Twice** during the semester, group members will be asked to fill out a detailed peer assessment for group members. The aggregate score (throwing out scores deviating more than 25% of the average) will become part of the student's grading score (10% of the project score - that's 4% of total score).

   Merely attending meetings won't be enough. Group members must be prepared for meetings, make good suggestions, perform their share of the work, and work well with other members.
   The grading criteria for peer assessment is as follows:
   a. Has the group member attended meetings?
   b. Has the group member been prepared for group meetings? I.e. was he/she aware of assignment requirements, performed her/his duties, able to speak intelligently about the project, etc.?
   c. Has the group member participated positively in meetings?