Overview

The CSCI 305 project involves the analysis, design, and implementation of a web-based database application based on material we have and will cover in class. The project is intended to be completed by pairs of students, though students may request and justify why they would like to work alone, and I may need to allow at least one group of three students in order to accommodate all students who want to work as part of a team. I will expect the effort put into the project to be commensurate with the number of people working on the project.

You may choose any application area that is of interest to you. However, the database component of your project should be substantial enough to support multiple ways of viewing the data, and should allow different types of users to undertake tasks that will support their roles. For example, a library system might allow library users to search for works, check out items, and check due dates for library materials in their possession; it would allow librarians to produce a list of overdue materials or determine who has possession of particular items.

Schedule

- **Wednesday, October 30:** Teams formed; I will have an email covering everyone’s plans. If you want a partner but don’t have one, please let me know as soon as possible.

- **Monday, November 4:** Project proposal handed in.

- **Wednesday, November 13:** Design phase 1 completed: Entity/Relation description of data handed in.

- **Wednesday, November 20:** Design phase 2 complete: relational schema, functional dependencies, and proofs of normal forms handed in.

- **Tuesday, November 26:** Initial MySQL implementation handed in.

- **Monday, December 9:** NOTE revised date. Full project including user interface and sample data completed and handed in.

Details

There is a wide range of possible domains that would be suitable for this project. The goal is to build a system for which it is possible to write interesting and useful queries from a number of points of view. In addition to the library system mentioned above, some other possibilities are as follows:
• An e-commerce application. Managers of the system should have the ability to add and delete items from the catalog. They should be able to determine the most popular items, to check stock levels for the products in the system, and to determine aggregate measurements such as net revenues or number of items sold. Customers should be able to search for specific items, assemble and place an order, and review their account history.

• A resource scheduling application. Consider managing something like classrooms or athletic facilities. Users who want to book facilities need to see which are available at times of interest, as well as see what properties various facilities have (capacity, equipment, etc). People managing the facilities need information about the schedule for various facilities (e.g., when is Dana 113 booked during the week of November 4?), as well as information about the people booking the facilities.

• A database covering some information-intensive topic such as the music industry, sports statistics, or collectible items. Such a system would let a researcher or enthusiast access the information in a wide variety of ways, and answer questions such as how many home runs Mickey Mantle hit in his second season, what the top-selling records were in February for each year in the 1980s, how many different versions of some Beany Baby were made, etc.

The tasks involved in the various project stages are as follows:

• **Project Proposal:** You will write up a proposal of between one and two single-spaced pages explaining your domain, why it is interesting and important, and why a database management system is needed. You should explain who the users of such a system will be and what tasks various types of users will undertake. You should give some idea of what data items you will need to store, and give examples (in English) of a range of queries you plan to support. You should also explain what you would like to provide in a user interface.

• **Design Phase 1:** You will develop a model of your database, including all entities, attributes of the entities, and relationships between entities. You will present your design as an Entity/Relation (E/R) diagram; this diagram will be done using diagramming software (see resources below).

• **Design Phase 2:** You will convert your E/R model into a relational database schema, along with a list of the functional dependencies that apply. You should prove that your database is in an appropriate normal form.

• **Initial MySQL Implementation:** Your MySQL database should contain all your tables, and you should have your tables populated with enough data to be able to test and demonstrate the key queries for your application. **Hand in** the MySQL statements you use to create your relations, and also for the main queries you support with your system. If you include assertions or triggers, hand in the MySQL code for that. Document the data you have included at this point in your relations (if your database is too large to do this reasonably, please let me know).

• **Full Project Completion:** Add a user interface to your system. You are free to use any high-level web programming language, such as PHP, Java/JSP, Javascript with Ajax, .NET, etc. Populate your database with sufficient data to provide interesting results that illustrate the operation.
of your queries. If you add indexes or views to your database, document what they are in your handin. Provide sufficient documentation so that I will be able to test your system.

Resources

There are many drawing tools available that let you draw diagrams, including xfig (available on our Linux systems) and various Mac and Microsoft tools. There are a few tools that I’ve seen recommended by other faculty members, though I haven’t used them myself. I’m not sure which are free or have free trials. Here are a few that have been recommended:

- Gliffy - http://www.gliffy.com
- Creately - http://creately.com
- MySQL Workbench

You can also search for “online diagramming tools” to find more.

Resources for developing user interfaces (to be provided):

- PHP
- JDBC (Java Database Connectivity)