CPSC350A Databases and Web Programming

Fall 2019

MWF: 9:40am--10:40am in Trexler 363

Dr. Anil M. Shende

Trexler 365A

Email: shende at roanoke dot edu

Tel: 540-375-2341

Office Hours: MW 11:00am-12:00noon; Th 2:00-3:30pm;

and by appointment.

Syllabus

Course Synopsis

In this course we will learn the theory behind relational database systems, and learn to design and create Webbased applications using a database at the back end. Topics include the architecture of a database system, the formalisms of relational algebra and relational calculus that form the theoretical framework for database systems, the query language SQL, normal forms of relations, and issues of data concurrency, security and integrity in the context of multi-user database systems. Through implementing a comprehensive Web-based application, we will learn about designing and creating interfaces for the user to meaningfully interact with the application, and the back-end, i.e., server-side, processing to support the application. One of the goals of the comprehensive application will be to design a database at the back end, and then enable Web-based data visualisation for large data sets stored in the database.

Required Texts

1. *An Introduction to Database Systems*, 8th edition by C. J. Date.

Prerequisites

CPSC 250 and MATH 131. Familiarity with Unix/Linux is assumed.

Intended Learning Outcomes

At the end of this course successful students will be able to:

1. understand the relational calculus underlying relational database design, and be able to formulate relational calculus statements to represent appropriate subsets of a given data set;
2. design normalized relational databases for a given data set;
3. use the Structured Query Language (SQL) to create relational databases and formulate queries to extract appropriate data from a given relational database;
4. use a variety of technologies to design user interfaces to be rendered on Web browsers;
5. design and implement a back-end for a Web application; and 6. use the graphical abilities of Web browsers for data visualisation.

Class Attendance

Regular attendance in class and the lab sessions is highly recommended. Regardless of attendance, students are responsible for all material covered or assigned in class.

Mechanics

The course will meet in class for three hours during the week. There will be two tests (on **Wednesday, September 25 and Wednesday, October 30**) in class during the semester. The comprehensive web application will take the place of a final exam.

This course expects you to spend *at least* 15 hours of work each week inside and outside of class.

Make-up tests will be available *by pre-arrangement only* in case of scheduling conflicts. After the test, make-ups will be available only in case of documented medical emergencies.

Besides the tests, there will be quizzes in class, regular homework assignments, and a co-curricular requirement.

Quizzes will be pop-quizzes, and will be at the beginning of class.

The tests and quizzes *may be **open handwritten notes**.*

***Comprehensive Web Application**: The assignment to complete the comprehensive application will have several parts:*

1. *initial mock-up design for the user interface,*
2. *detailed specifications for the services provided by the application, and thus the data that needs to be obtained from the user, displayed to the user and stored at the back end,*
3. *identification of noteworthy stages in the development process, and a clear timeline for the completion of these stages,*
4. *an assessment rubric for evaluating the progress at each stage,*
5. *a final report that clearly describes the use of the application (i.e., a user's manual), and describes the stucture of the developed application (i.e., a technical/developer's manual),*
6. *and a final presentation demonstrating the use of the application.*

***Co-curricular Requirement**: The Mathematics, Computer Science and Physics department offers a series of discussions that appeal to a broad range of interests related to these fields of study. These co-curricular sessions will engage the community to think about ongoing research, novel applications and other issues that face these disciplines. Each student is required to attend at least three of these sessions, and turn in a short paper describing the contents of the session, and his/her critical reflections about the topic and content. These papers are due in class within a week of the session. A paper submitted beyond a week from the event being discussed in the paper will not be accepted.*

Grading

The final grade will be computed based on the grades in the tests, the final exam, home works and programming projects according to the following weights.

Component	*Weight*	
Co-curricular	*4%*	
Home works	*15%*	
Quizzes	*15%*	
Tests (2)	*26%*	*(13% each)*
Web Application	*40%*	

The final course grade will be calculated as follows:

< 60	*60-62*	*63-65*	*66-69*	*70-72*	*73-75*	*76-79*	*80-82*	*83-85*	*86-89*	*90-92*	*> 92*
F	*D-*	*D*	*D+*	*C-*	*C*	*C+*	*B-*	*B*	*B+*	*A-*	*A*

Class Attendance and Policies

Regular attendance in class is highly recommended. Regardless of attendance, students are responsible for all material covered or assigned in class.

Cell phones should be kept in your backpacks or pockets (essentially, out of sight), and turned to silent mode throughout the duration of the class. Please do not remove your cell phones until you are outside the classroom/lab. Similarly, during office consultations or consultations in the lab (even when it is not during regular class time), your cell phones should be out of sight and in silent mode.

*Note above that the tests and quizzes may be open **handwritten notes**. You will not be allowed to take photos of the board.*

If you use an electronic device such as a tablet or a laptop for note-taking or to read the textbook, the content that is open on the screen should be strictly restricted to documents and pages of relevance to the class. For example, you should not have any social media websites open in your browser window, even if it is in a tab that is not currently in focus.

Academic Integrity

Students are expected to adhere to the Academic Integrity policies of Roanoke College. All work submitted for a grade is to be strictly the work of the student unless otherwise specified by the instructor. The policies as outlined in the Academic Integrity handbook will be enforced in the course.

Graded programs are subject to the Roanoke College Academic Integrity policies. Copying a program or a portion of a program (even a single line) or reading another person's program to obtain ideas for solving a problem is plagiarism. Other examples of integrity violation include writing code for someone else, using code written by someone else, telling someone else how to solve a problem or having someone tell you how to solve a problem (and using his/her method). These cases apply to any work that is handed in for a grade under the instructor's assumption that the work is your own. Unless specified otherwise by the instructor, discussion among students should be limited to general discussion of concepts and language details, not specific aspects of a solution to the assigned problem.