

# INFSCI 2710: Database Management

## PROJECT: Database System for E-Commerce

## Purpose of the project

You are to analyze the requirements for, design, implement, document and demonstrate a database system that could automate the functions of a sales company with E-Commerce.

## Application Requirements

You should select specific products for your project to sale and describe specific application requirements. You should implement a database where your system must maintain a current record of inventory that is updated to reflect the transactions performed. You must keep track of customers, including their personal information, and purchase history. A catalog of information about products for sale must be maintained in a way that is searchable by customers.

All projects are required to demonstrate *at least* the following functionality:

### ***Data***

- **Customers:** customer ID, name, address (street, city, state, zip code), kind (home/business). If business, then business category, company gross annual income, etc. If home, then marriage status, gender, age, income.
- **Products:** product ID, name, inventory amount, price, product kind w.r.t. some classification.
- **Transactions:** record of product purchased, including order number, date, salesperson name, product information (price, quantity, etc.), customer information.
- **Salespersons:** name, address, e-mail, job title, store assigned, salary.
- **Store:** store ID, address, manager, number of salespersons, region.
- **Region:** region ID, region name, region manager.

### ***Operations***

- **Customer Browsing** The users must be able to search the operational database for particular items based on various attributes and must also be able to do browsing (i.e., less focused searching).

- **Update Transactions** The system must be able to handle payments and sales, new inventory, new users, etc. and other changes to the operational database that are necessary for the day-to-day running of the business.
- **Error Checking** The system must be robust and support various application-dependent integrity constraints. For example, items should not be sold if they are not currently in stock, etc.
- **Data Aggregation** The system must provide data aggregation queries:
  - What are the aggregate sales and profit of the products.
  - What are the top product categories.
  - How do the various regions compare by sales volume?
  - Which businesses are buying given products the most?
  - Other interesting aggregate queries that you will come up with.

## Rules of the game

- **Implementation Tools:** All projects are expected to be runnable from a web browser from the instructor's office. SQL is a requirement. A group may choose to use any database systems and front-end implementation tools after discussing it with instructor.
- **Additional Requirements:** The project must represent a fairly sophisticated database application. In particular, the database must contain multiple (e.g., at least seven) relations, and the database design must include indexes, primary keys, etc.
- **Groups:** The project is to be done in groups of 3 students. A roster for each group must be submitted to the instructor by the date specified in the "Due Dates" section of this assignment. The groups are ``self-policing'' (e.g., each group is responsible for its own division of labor, scheduling, etc.). *Note: If an unreconcilable problem arises in your group, it is your responsibility to contact the instructor as soon as possible. After the project is due, it will be too late.*
- **Assumptions:** In cases where the above description of the application is incomplete, it is acceptable to make assumptions about the application providing that: 1) they are explicitly stated in the final report, 2) they don't conflict with any of the requirements specified above, and 3) they are "reasonable". If you have a question about the acceptability of any of your assumptions, check with the instructor or GSA. Interesting questions should be raised in class.
- **Report:** A final report should be handed in for grading at the end of the term. The report must be formatted in a reasonable manner (i.e., using a text processor and a decent printer). The final report is due during class on the "Project Due" date specified in the class schedule.
- **Implementation:** The project requires a working implementation of the system to be built, tested, and demonstrated. A large part of the project grade depends on the quality of this implementation.

## Report Requirements

*The final report must contain:*

1. A short overview of the system including identification of the various types of users, administrators, etc. who will be accessing the system in various ways.
2. A list of assumptions that you have made about the system.
3. A graphical schema of the database using the E-R diagram with a short description of each entity set, relationship set and their corresponding attributes.
4. A set of relational schema resulting from the E-R diagram with identification of primary and foreign keys.
5. The DDL statements to create the relational schema in some appropriate Normal Form, with identification and justification of the Normal Form.
6. A description of your front-end design as well as the front-end to back-end connection.
7. A brief overview of the system implementation with example screen shots.
8. A description of your testing efforts and erroneous cases that your system can detect and handle.
9. A description of the system's limitations and the possibilities for improvements.

**In addition, a demo of the working system will be required. All members of the group must attend this demo, and must be prepared to explain and demonstrate those aspects of the project for which they were responsible. The source code for the project should be available on-line during the demonstration.**

**Demos will be scheduled after the last class and before the final exam.**