CSE 3241/5241: Database Design and Implementation Project

The purpose of this project is to give you some experience in the database design and implementation. We will explore both theoretical and practical aspects. Once you master it well, it is a simple matter of translating the theoretical concepts you developed into a working database.

This is a team project. Up to three students constitute a team. You (as a team) must submit copies of your final report (project description, ER model, queries in algebra, relational tables, and SQL scripts (schema creation to include tables/keys/constraints, data inserts, constraint checks, and query implementation) and schedule a 10 minute demo session with the grader to demonstrate the functionality of the software (no grade will be assigned for the project without the software demonstration).

Project Phase 1: Database Design

Description: The first phase of the Project consists of the following tasks:

1. Collect and analyze data management requirements of an organization of your choice (make sure you have sufficient domain knowledge). Sample organizations include an airport, grocery store, car dealership, restaurant business, hotel business, software development company, health care organization, etc. Provide the description of your organization and its operations. Below is a sample description for a typical banking enterprise.

A bank normally has many branches, and customers can open accounts at any of these branches. It is normal for more than one customer to have the same account and for one customer to have multiple accounts. Each customer must have at least one account and each account must have at least one holder. The bank offers four types of accounts: (1) checking, (2) savings, (3) credit, and (4) retirement. The bank issues a debit/credit card for each customer with a checking/credit account. Each retirement account has a financial advisor who can manage multiple retirement accounts. The bank records all the transactions for each account, including deposits, withdrawals, payments, balance transfers, late fee and financial charges, etc. Some additional requirements and information may include primary and secondary holders for an account, online banking, mutual funds for retirement accounts, interest rates, etc.

You should provide similar description of your organization based on your analysis.

2. Design an ER diagram that captures the information about your organization and its operations. The diagram should contain at least 15 entities with their associated relationships. Show attributes for each entity and relationship type. Make sure that your ER model captures the requirements described in the previous task.

3. Indicate any key, participation, and cardinality constraints on the diagram.

4. Document all the data integrity constraints and non-trivial decisions you make while modeling your organization.

For example, for the banking enterprise, one can choose a SSN to be a primary key of entity type CUSTOMER. Such a decision, although seems simple and natural, has some serious consequences: the bank will not be able to serve clients who have no SSNs, which may not always be true in real life. You should be able to recognize and document this and similar decisions for your design and state appropriate assumptions. An example integrity constraint may state that a customer cannot open credit account if she/he does not have a savings account with the bank. This constraint will not appear on the diagram and will require

a trigger to be implemented later in the project.

5. Translate your final ER diagram to the relational tables.

For example, for the banking enterprise, the design may result in the following relations:

CUSTOMER(SSN, Name, DOB, Address, Phone, …)
TRANSACTION(…)
etc.

Submission: Submit your project description, ER diagram and relational tables.

Project Phase 2: Database Schema Implementation

Description: The second phase of the Project consists of the following tasks:

1. Create an SQL scripts for Microsoft SQL Server Express that generate the database schema designed in the first phase of the Project. To enforce data integrity constraints, use `PRIMARY KEY`, `FOREIGN KEY`, `UNIQUE`, `NOT NULL`, and `CHECK` constructs within SQL's `CREATE TABLE` statement.

2. Add at least two triggers in SQL script to enforce additional data integrity constraints defined in the database design report (you may add new constraints to the report if needed).

3. Describe indexes that were automatically created ("side effects") by Microsoft SQL Server Express on table columns that were specified as primary keys in the `CREATE TABLE` statements. Add additional indexes to support efficient evaluation of queries that involve other table columns.

For example, for the banking enterprise, a SSN can serve as a primary key of table CUSTOMER and will have the corresponding index (on the SSN column of the table). However, some queries will search customers based on their FirstName and LastName. For such queries, an index on the list of columns (FirstName, LastName) or (LastName, FirstName) may be helpful.

4. Add `INSERT` statements to the SQL scripts to populate each table with a few rows.

5. Add comments to the SQL scripts to describe your implementation (e.g., how your indexes are helpful, how you tested triggers, etc.).

Execute and test your SQL scripts in Microsoft SQL Server Express. Pay special attention to the triggers; make sure that they indeed enforce constraints (you may need to use `INSERT`, `DELETE`, `UPDATE`, and `SELECT` statements to check how they work).

Submission: Submit your revised project report (project description with assumptions and constraints, ER diagram, and relational tables) and SQL script file (for database creation & data load).

Project Phase 3: Database Query Development & Implementation

Description: The third phase of the Project consists of the following tasks:

1. Create the database schema in Microsoft SQL Server Express using the previously developed SQL scripts; populate the database with sample tuples, such that the result of the queries described later is not empty;

2. Write the following queries in algebra and SQL for your database that use (1) union, (2) intersection, (3) difference, (4) division, (5) aggregation, and (6) at least three joins (inner or outer).

For example, for the banking enterprise, sample queries are as follows.

Query 1: Retrieve information about customers (e.g., SSN/ID and name) who have checking or savings accounts. [Hint: union]

Query 2: Retrieve information about customers who have both checking and savings accounts. [Hint: intersection]

Query 3: Retrieve information about customers who have both checking and savings accounts, but not a retirement account. [Hint: intersection and difference]

Query 4: Retrieve information about customers who have all types of accounts. [Hint: division]

Query 5: Retrieve information about each account (e.g., account number) and (next to it) number of recorded transactions for the account. [Hint: aggregation]

Query 6: Retrieve names of customers and their financial advisors. [Hint: inner joins between Customer, Account, Manages, and Advisor]

3. Implement the above queries in SQL, allow users to execute the queries in the database and return its result to the user.

Submission: Submit your revised project report (project description with assumptions and Constraints, ER diagram, and relational tables), SQL script file (for database creation & data load, and SQL queries) and algebra queries.

Project Phase 4: Demo

Submission: Present a 10 minute demo to the instructor or grader during the class on a specified date to demonstrate the functionality (no grade will be assigned for the project without the demonstration).

Project Phase 5 (Optional – 25% of project grade): Implement the above queries in SQL (use project phase 3) and allow users to view and execute the queries from a website, and see the results on web pages.

Submission: Present a 5 minute demo to the instructor or grader during the class on a specified date to demonstrate the functionality (no points will be given without the demonstration).