Library Management System

Georgia State University is interested in building a robust web-based application for managing its books, journals and library users. The CIO of Georgia State clearly understands that a good foundation is necessary to build a good application. He wants your team to build an appropriate database schema for the web-based library management system and demonstrate a basic implementation in SQL Server 2005 based on the schema. He did some research and created a requirements specification for the system.

(Please note that most requirements were gathered from the application perspective and the database designers are expected to understand the application and design the schema for the application.)

Requirements Specification

User Management

The system should have three types of users – student, faculty and administrator. Every student can borrow two books and can hold them for a maximum of 25 days. For every day after the deadline, the student is expected to pay a fine of $1 per book per day. The student is expected to pay 15% of the price of the book along with the actual price of the book once he/she crosses the 50 days limit beyond the day of borrowing. Every student is allowed to reserve two books at any time. Their reservation is valid for 1 day from the time the book reaches the library shelf. The student and faculty can reserve an already reserved book. The users will be notified via email when the book becomes available. Every faculty can borrow five books at any time and can hold them for a maximum of 45 days. For each borrowed book the faculty members are expected to pay the price of the book once they cross the 100 days limit from the day of borrowing. Every faculty can reserve a maximum of 5 books. The faculty members’ reservation is valid for 3 days from the time the book reaches the library shelf. The administrator is not an actual user and he should be given enough privileges to change the parameters specified above (the number of books a student/faculty can borrow, the number of days they can hold them, etc).

Please capture the following information for the users:

1. First Name
2. Last Name
3. Panther ID
4. Gender
5. Address
6. Degree (Undergraduate/Masters) – For students
7. Major – For Students
8. Department – For faculty
9. Phone number
10. Email
Application level expectations

The web application developers are not really database specialists and they would like the database designers to provide them the required queries. Please provide the following queries:

1. Get the number of total users, students alone and faculty alone
2. Get the complete details for any specific user
3. A query to get the parameters (the number of books a student/faculty can borrow, the number of days they can hold them etc) for a user type

Books Management

The application should be capable of capturing the following details for each book:

1. ISBN
2. Title
3. Author (First, Last, Middle Name)
4. Quantity
5. Cost
6. Category (Fiction, Science etc)
7. Publisher
8. Date of publication
9. Language of publication

The users should be able to search for the books based on ISBN, Category, Author, Category, Keywords, publisher etc. Any book can belong to multiple categories. Any book can have multiple authors (but only one can be the primary author).

Application level requirements

Please provide the following queries and stored procedures (as a database designer, you can decide which ones should be queries and which ones should be stored procedures):

1. The number of the unique books the library has
2. The number of books in each category
3. All the details of a particular book given the ISBN
4. All the books written by a particular author (the query should accept last name or first name or both)

Library Management System

The users can write reviews about any book and can rate them in a scale of 1 to 5 (5 being the best).
Please provide the following queries:

1. A query to retrieve the fine money collected in a specific period of time
2. A query to retrieve the ISBN of the books that needs to be purchased in a specific period of time (based on the books that has not been returned and for which the student/faculty is expected to pay the price for the book)
3. A query to retrieve the top 10 books which have been overbooked in a specific period of time (so that the librarian can buy some more copies)
4. A query to retrieve the list of books which have never been borrowed
5. The list of books borrowed by a specific user
6. A query to calculate the money a user should pay (both fine and cost + 15% of cost of the book - depends on user type)
7. A query to calculate the average rating for any book

Please see the “Database Management System Project” document for other general requirements for the project.