National University of Singapore
School of Computing

BT2102 Data Management and Visualisation

Assignment 1 (Group Project):
A Library System (ALS)

Total marks: 40

NARRATIVE

A library system (ALS) is required to keep records of books and provide services for library members to borrow, return, and reserve books. The library is not a typical library that you may be aware of. It only has 50 books available for loan. It intends to increase the number of books for circulation in twelve months' time, but you are told this is not a concern for this project.

To facilitate the construction of ALS, the following **functions** have been gathered from the library administrators:

1. **Membership Creation**
   - To be able to use the library facilities (such as borrowing or reserving books), an individual must become a library member.
   - Information required to be a member includes membership id (a unique alphanumeric id that distinguishes every member e.g., A101A, A901I, etc.), name (e.g., Tan Jia Yu, Joan Lim, Simon Li Thales, etc.), faculty (e.g., Computing, Engineering, Science, etc.), phone number (e.g., 91234567, 81093487, 92054981, etc.), email address (e.g., tanjiayu@als.edu, jlim@als.edu, etc.).
   - A member can borrow (i.e., number of books borrowable) a maximum of 2 books at any time.
   - The membership id is assigned when a membership is created.
   - Members are distinguished by membership id and not by name i.e., two member records with the same name "Tan Jia Yu" are considered distinct if the membership ids are different, and two member records are the same even with different member names if the membership ids are the same.
   - The database cannot have two member records with the same membership id.

2. **Membership Deletion**
   - An individual membership with the library can be terminated at any time.
   - When this happens, the membership record is deleted.
   - All books loaned to members must be returned prior to the deletion.
   - All outstanding fines must be paid prior to the deletion.
   - All reservations requested by members will be cancelled.

3. **Membership Update**
   - Membership record details can be updated at any time.
   - Information that can be updated include member's name, faculty, phone number, and email address.
   - Membership id, the number of books borrowable, and the number of books reservable cannot be changed.

4. **Book Acquisition**
   - Library acquires new books for use by library members.
   - Each book has the following details: accession number (used to identify an instance of book), title, authors (there can be multiple authors for a book), isbn, publisher, and publication year.
   - All books have unique accession number.
   - Books may have the same isbn but not the same accession number since there can be multiple copies of the same text.

5. **Book Withdrawal**
   - Library books that are outdated will be withdrawn from circulation.
   - The book record is deleted from the database.

- Only books that are currently not borrowed can be withdrawn from circulation.
- Any book on reservation can be withdrawn after the book has been removed from all reservations.

6. Book Borrowing
- A book can be borrowed if it is not on reservation or already on loan to another member.
- Each book loan is valid for 2 weeks. The due date is therefore a date 14 days after the borrow date (or loan date). When the 14 days are up, the book on loan must be returned. There is no renewal service.
- A member can borrow a book if he/she does not have an outstanding fine (i.e., the fine has not been paid).
- Information captured for a book loan includes accession number, borrow date (taken from the system), due date (a derived attribute), and membership id.

7. Book Returning
- A book on loan must be returned to the library on the due date.
- A member has a fine when a book is returned later than the due date. A penalty of $1 per book per day is imposed as a fine to the member.
- The fine amount is cumulative if more than one book is returned late.
- A book returned can be borrowed by the same or another member.
- Information captured for a book return includes accession number and return date.

8. Book Reservation
- If a book is currently not available (because it is on loan to a member), a member can request for the book to be reserved.
- When a reserved book becomes available (because it has been returned), it can be loaned only to the member who reserved it. When the reservation is fulfilled (i.e., the book is loaned to the member who reserved it), the reservation is deleted.
- A member can only reserve a book if he/she does not have an outstanding fine.
- A member can reserve (i.e., number of books reservable) a maximum of 2 books at any time.
- Information captured for a book reservation includes accession number, reserve date, and membership id.

9. Reservation Cancellation
- A reservation on a book on loan can be cancelled at any time before the book is available for loan.
- When it is cancelled, the reservation record is deleted.
- Information captured for cancelling a book reservation includes accession number and membership id.

10. Book Search
- A member can perform a basic search on the collection of books. No sophistication on the search is required.
- Search can be done by specifying ONE word in the title, authors, isbn, publisher, publication year field.
- A searchable word is a single distinct element of writing e.g., "man" is a word but "man" in "batman" is not a word; however, "batman" is a word.
- The search field will only accept one word for each search and on any of the book attributes e.g., a search for word like "Batman" on title, "2011" on publication year, "Isaac" on author, etc. It does not take in more than one word for any search field/attribute.
- The search results will display information on the book including accession number, title, authors, isbn, publisher, and publication year.

11. Fine Payment
- Fines must be completely paid. No partial payment is allowed. For example, if the fine amount is $4, only $4 payment is allowed; any payment less than or more than $4 will be rejected.
- Information captured for fine payment includes ~~accession number~~, member id, payment date, payment amount.
- This function does not facilitate the actual collection of the payment. It merely records the payment amount, date and indicate the fine has been fully paid e.g., update the fine amount to 0.
- No further action is required after the fine has been paid.

12. Display books on loan
- This function displays all the books currently on loan to members.
- The information on the books displayed include accession number, title, authors, isbn, publisher, and publication year.

13. Display books on reservation
- This function displays all books on reservation to members.
- The information on the reservation displayed include accession number, title, membership id, name.

14. Display members who have outstanding fines
- This function displays all members who have outstanding fines.
- The information on the members displayed include membership id, name, faculty, phone number, and email address.

15. Display the books on loan to a member given the membership id
- This function displays all the books a member identified by the membership id has borrowed.
- The information on the books displayed include accession number, title, authors, isbn, publisher, and publication year.

ONLY MySQL database will be used to store all information on books, members, loans (book borrowing and returning), reservations, fines, and payments on fines.

WHAT YOUR TEAM IS REQUIRED TO DO?

1. **Produce a Powerpoint (or similar) presentation** on the development of a conceptual data model and logical data model for the relational database based on the above **Narrative**. The relational database will be implemented on a **MySQL** database (**Do not use SQLite**).
   To produce the conceptual data model, a summary of the steps is given below:
   a. Identify entities, their associated attributes, and primary key attribute(s).
   b. Identify the relationships between entities and the associated relationship attributes and cardinality (including multiplicity and participation constraints).
   c. Produce an Entity-Relationship Diagram depicting the conceptual data model.
   d. Transform the Entity-Relationship Diagram into relational schemas (i.e., tables).
   e. Refine the relational schemas by combining schemas and normalize the tables where necessary to produce the **final logical data model**.

2. **Implement the relational database on MySQL** (using the MySQL Workbench) based on the final logical data model:
   a. Produce SQL scripts to create tables and specify all essential constraints via MySQL Workbench.
   b. An initial set of 50 books together with their attributes are given in the **libBooks.txt** file.
   c. An initial set of 9 members together with their attributes are given in the **libMems.txt** file.
   d. Initialize the database with these 50 books and 9 members i.e., there will be 50 books and 9 members inserted into the Book and LibMem tables (You may use MySQL workbench to directly insert these records).

3. **Implement all the functions** (as described in the **Narrative**) in a software application to facilitate the operation of ALS. The functions will interface with the relational database to produce the required results:
   a. Develop the ALS functions in **Python or Java only**. You may use any of the interface framework/middleware available in Python or Java (e.g., Object Relational Mapping (ORM) in Python (such as SQLAlchemy) or Java (such as Hibernate, TopLink)) to connect/interface with the MySQL database. You may also use pymysql as the connection interface between Python and MySQL database.
   b. ALS will be developed as a standalone application. **You are not required and should not develop ALS as a web-based or mobile application**.
   c. Follow the user interface design as shown in the **ALS-UserInterface.pdf** file for the various functions in ALS. For a standardized user interface design, use **TKinter for Python** or **Swing for Java** graphical user interface design. **No other GUI tools are required or allowed**.
   d. **There is no requirement for any login page to start and use ALS.**

4. **Project consultation**:
   a. Due to the large class size, project consultation will be assigned to the Graduate Teaching Assistants (GTA). See "**Project Consultation Assignment**" section below for the allocation of GTA to project groups. Contact/email the respective GTA should your group requires consultation.
   b. Project consultation is limited to the **clarification of functions, conceptual/logical database design and MySQL scripting concepts**. For example, questions on whether your Conceptual or Logical Design model is correct according to stated Narrative **will not be answered**; neither will there be answers on whether a MySQL database has been scripted correctly for your application.

You should have learnt how to produce Conceptual or Logical Data model, and MySQL scripts in the tutorials/lectures.

   c. No consultation will be provided on how to write Python or Java code for the development of ALS. You are expected to learn and manage the production of the ALS system as a team.

   d. **All students taking this course are expected to know how to produce Python or Java code**.

DELIVERABLES

1. A **PowerPoint** (or similar) presentation of the conceptual and logical data model. Only the following need to be included in the PowerPoint file: Final Entities and Attributes, Final Conceptual Data Model, Final Logical Schema, and Final Logical Data Model. A sample PowerPoint file ("**BT2102-A1-Sample.pdf**") has been provided as a guide to the production of this deliverable. **Maximum number of Powerpoint slides is 8 slides. 5 marks will be deducted for any PowerPoint submission that exceeds 8 slides.**

2. A demonstrable ALS application that can achieve all the required functions stated in the **Narrative**.

3. All codes (MySQL, and Python or Java) used in this group assignment zipped into single file.

SUBMISSION

1. Zip the PowerPoint file, together with the MySQL, and Python or Java zip file into a single zip file and name the zip file as "**GRP_XX_AS1.zip**" (for example, file name for group 9 is "GRP_09_AS1.zip" and for group 19 it is "GRP_19_AS1.zip"). **5 marks will be deducted for non-conformance to file name.**

2. Upload the zip file to "Assignment 1 Submission" folder in LumiNUS->Files by the assignment due date.

3. **THERE IS NO REQUIREMENT FOR YOUR GROUP TO MAKE A PRESENTATION OF THE POWERPOINT SLIDES, BUT A FACE-TO-FACE OR ZOOM DEMONSTRATION WILL BE ARRANGED INDIVIDUALLY FOR EACH GROUP IN WEEK 10 OR 11 TO EVALUATE THE FUNCTIONAL EXECUTION OF THE ALS APPLICATION.**

4. Please be reminded that **plagiarism is a serious offence**. Disciplinary actions will be taken against those caught cheating.

EVALUATION

1. Your project will be evaluated as follows:
   a. The correct development of the conceptual and logical data model. You will not be required to present the PowerPoint slides. (**6 marks**)
   b. The ALS functions as listed in the **Narrative** (evaluated in a face-to-face or zoom demonstration to be held individually for each group in Week 10 or 11). (**30 marks**)
   c. The ALS application user interface (as described in "**WHAT YOUR TEAM IS REQUIRED TO DO?**" section 3c) to facilitate the functions in the **Narrative**. (**2 marks**)
   d. The ALS application interface connection with the MySQL database. (**2 marks**)

PROJECT CONSULTATION ASSIGNMENT

Group Number	Graduate Teaching Assistant Assigned	Email Address
01 to 13	Mr Alvaro Gonzalez Rivas	alvaro@u.nus.edu
14 to 26	Ms Kong Guoting	e0546217@u.nus.edu
27 to 39	Mr Liu Yiming	liu_yiming@u.nus.edu
40 to 52	Ms Qi Qi	qi.q@u.nus.edu
53 to 65	Ms Wang Yuchen	yuchen.wang@u.nus.edu
66 to 78	Mr Zhang Yifan	yifan.zhang@u.nus.edu