Assessment Brief: Coursework 2023-24

Assessment Details

Course Title:	Database Management Systems
Course Code:	LCSCI7228
Course Leader:	Ibukun Afolabi
Level:	7
Assessment Title:	Project
Assessment Number:	AE1
Assessment Type:	Documentation, Code, Video
Restrictions on Time/Length:	Code and up to 2000-word documentation
Individual/Group:	Groups of 1, 2 or 3
Assessment Weighting:	40%
Issue Date:	17 May 2024
Hand in Deadline:	5 July 2024, 13:00
Planned Feedback Deadline:	28 calendar days after hand in deadline
Mode of Submission:	Online
Anonymous Marking:	No

Assessment Task

You have been hired as Lead Database Engineer for a University Library Management Application. The goal of this application is to provide an efficient process for students to have access to books and other resources that the library offers. When students request to borrow books, it goes through a request process and if their request process is successful, the book is released to the student on loan. The application is looking to replace a clumsy and manual books management system that students have complained about. The book lending process should be an easy and efficient way for intending students to enlist and borrow books and resources. Books are in both digital(softcopy) and non-digital(softcopy) versions. The application should not charge for borrowing books (hard copy) but if the borrower (students) keeps the book for more than 7 days then the student should be charged a little amount. Develop an application with the following modules:

User Management. Users provide minimal information upon registration: first/last name, email address (used for login/notifications), a password, a profile picture, contact phone number(s), and a contact address (street, city, postal code, country). User's can take on one or more roles: Students, Staff, librarian, and/or administrator. Users are allowed to borrow more than one book or resource materials. Librarians can log into the system and review the book request. The librarian should grant the request if all criteria are met by the borrower. If the request is not granted then there should be a feedback to the borrower. If a request is granted, hard copy cooks can be picked up or posted to the borrower after a token is paid. The book request process requests the user's title, affiliation, and interests. Administrators have a variety of enhanced positions within the system, and must thus be granted their role by another administrator – for purposes of security auditing, the date/time and grantor must be recorded.

Books and Resources. Each book/resource has a title, author(s), description, and publisher. Books are categorised by title – each book has a primary title, as well as any number of secondary titles. Users can borrow more than one book. If a user borrows a book, the date/time must be recorded. Additionally, once the book is returned users provide a rating: 1–5 stars of the book, as well as any comments they might have for improvement/potential authors.

Criteria to borrow books/ resources. To borrow a book, you must have returned the previous one(s) you borrowed. You need to pay all outstanding bills if any. For students they must be currently registered in the university, Outgone students cannot borrow books.

Sales The library has a small sales section where excess and old books are books/resources can be sold. The librarian should be able to monitor the sales of the books and regulate the book and resources such that books meant to be borrowed are not sold. Only books that have excess copies in the library are sold. Only the librarian can sell books/resources.

Different users are allowed different levels of access to the system based on their user privileges. Users should have the opportunity of asking for feedback and getting regular updates concerning borrowed books or books available for sale.

Your application is also required to include the following tasks

a) Register a new user
b) As an administrator, authenticate a user (based upon email) or fellow administrator.
c) Provide a categorised list of available books or resources (each with primary/secondary topics, ranked by average evaluation score or review score).
d) Enrol a user to borrow or buy books or resources

e) For an enrolled user, request to borrow or buy library materials
f) Librarians should be able to request for the overall state of available resources in the library.
g) Librarians should be able to categorise the book/resources to be borrowed or sold.

Assessment Criteria

The goal of the project is to gain hands-on experience with developing a full database application from the ground up. Given a narrative, the students will form a group, design a set of relations, populate the database, write a set of contextually useful SQL queries, and then implement a user interface to execute these (parameterized) queries, and view results, in a programming language/stack of their choosing.

Students will work in project teams, typically with 1, 2 or 3 members (grouping will be done by the course leader).

Individual Submission

- Report (report.pdf), which includes:
    - Abstract: a 1-page overview of the problem being solved and the solution delivered (½ page)
    - A detailed textual description of the problem, including what entities are involved, how they relate/constrain each other, and what information needs to be extracted/presented – this is your opportunity to make concrete any assumptions you had to make based upon ambiguities in the supplied narrative(1page)
    - ER diagrams: 1 global, as many logically separated local views as is appropriate
    - Normalised relations with primary/foreign keys clearly identified; any deviations from the ERD and/or 3NF must be justified in text (including denormalization)
    - Physical design: including justification of any included indexes, denormalization, etc.
    - Screenshots of your running system, demonstrating example input/output of each task
    - A project retrospective, including at least(1 page):
        - what you liked and disliked the most
        - what you found the easiest and the hardest

            - what you learned from doing a large-scale project
            - What recommendation would you give to scale the database in future.
        - § A conclusion statement: what was produced, what still needs to be done (½ page)
    - A link to a **private GitHub source code repository** (repo.txt), which includes:
    - Structure and data dumps (i.e. DDL & DML)
    - Application source code
    - A README.md file in the root indicating all of the program names, what they do, and how they are compiled, configured, or used.

Group Submission

- Presentation. The presentation is intended to be a brief overview of the project, including design and implementation. All members of the group are required to visibly and/or audibly contribute to the group presentation; if a group member is not clearly presenting, s/he will receive a reduced individual project grade. The presentation will be made using PowerPoint (or equivalent) and must cover at least the following aspects of the project:
    - Project Motivation: non-technical overview of the problem being solved
    - System Description: what does the system do? How does this solve the problem? baseline features (i.e. tasks), as well as any additional above-and-beyond functionality your team added
    - System Architecture: what DBMS front-end language(s), and any other component(s)? Why? How does everything talk?
    - Database Design: ERD, relational model, source of the data for each table, physical design Your presentation should last 8–10 minutes.
- A link to a **private GitHub source code repository** (repo.txt), which includes:
        - Structure and data dumps (i.e. DDL & DML)
        - Application source code
        - A README.md file in the root indicating all of the program names, what they do, and how they are compiled, configured, or used.

Submissions will be graded for quality and completeness using the following criteria:

1. ERD, Mapping - A detailed textual description (description.pdf) of the problem, including what entities are involved, how they relate/constrain each other, and what information needs to be extracted/presented. Include any assumptions you had to make based upon ambiguities in the supplied narrative. ER diagrams (erd.pdf): 1 global, as many logically separated local views as is

appropriate. Normalised relations (logical.pdf) with primary/foreign keys clearly identified; any deviations from the ERD and/or 3NF must be justified in text (including denormalization). DDL (ddl.sql): SQL that instantiates your normalised relations in a DBMS of your choosing (must be identified in the comments)
2. Data, Queries - DDL (ddl.sql): possibly updated from Milestone. DML (dml.sql): SQL that populates your tables (each relation should have at least 10 rows, moreso as necessary to demonstrate tasks/reports). Task Queries (tasks.sql): SQL that implements each of the required tasks (provide comments where appropriate for parameterization, sequencing, etc). Report Queries (reports.sql): SQL that implements each of the 5 complex reports (provide comments describing the purpose, as well as justification of the complexity).

Report Complexity - You are to write three classes of queries for this project:

1. Queries for your system to install/run (e.g. creating tables, populating tables, and simple queries to add/update/remove entities)
2. One or more queries to implement each of the required narrative tasks
3. 5 complex reports , select from the list below:

- Tables joined (Non-inner/natural join?)
- Subqueries
- Aggregate function(s)?
- Grouping?
- Ordering fields
- WHERE/HAVING conditions
- Non-aggregation functions or expressions in SELECT/WHERE?

Marking

The total grade of 40 marks is based upon…

Individual project grade **(20 marks (**50%)**)**

1. Report (**15 marks**). Includes the sample data (quantity, quality, strategy and implementation for gathering/producing the data), has quality (did it have all required content, was the writing grammatically correct and professionally presented). Correct and efficient implementation of the supplied tasks, as well as context-appropriate design & correct/efficient implementation of additional 5 report queries. The submitted report will be assessed for English proficiency (**4** out of the **15 marks**).

2. The source code **(5 marks)** was it commented; were user inputs sanitised; were good security measures taken, such as salting passwords), and the ability to install, run, and use the system. Note that a “pretty” and usable interface is encouraged, and may be given bonus marks for outstanding work, but is not required (i.e. if command-line tools yield a fully functional system, all associated marks will be awarded).

The group project grade **(20 Marks** (50%)**)** is based upon . . .

1. Presentation (**10 Marks)** Includes aspects of professionalism (how the group presented itself, was the presentation rehearsed, did it fit well into the time allowed, production quality) and the degree to which the required sections were covered (see above).
2. Database Design (**10 Marks**) the presentation includes an explanation of the ERD and resulting relational schema (was the design correct semantically, as well as technically both in diagrams and code; were tables normalised to at least 3NF; was a reasonable physical design applied and justified, relative to queries).

Learning Outcomes

This assessment will enable students to demonstrate in full or in part the learning outcomes identified in the Course Descriptor.

On successful completion of this assessment, students should be able to:

Knowledge and Understanding

K1d Master practical tools, methods and techniques required to build a database application.

K2d Understand advanced aspects of database design theory, query language, and performance/tuning issues of databases.

K3d Evaluate relational database management systems as a class of software systems, and their technical, social and management dimensions when deployed in multi-user environments.

Subject-Specific Skills

S1d Critically assess a data problem, recognise the individual tools, libraries and techniques suitable for solving that problem and integrate them to produce a database application.

S2d Become a sophisticated user of database management systems and develop software that integrates available database systems using best industry practices and standards.

S3d Design and develop original software for a database application that solves a practical data problem.

Transferable Skills

T1d Critically review existing database technologies and propose the right tools for solving a problem or ways to improve it.

T2d Communicate effectively the design and implementation choices of a database application.

T2d Consistently display an excellent level of technical proficiency in written English and command of scholarly terminology, so as to be able to deal with complex issues in a sophisticated and systematic way.

T3d Lead or participate in team projects.

Accessing Feedback

Students can expect to receive feedback on all summative coursework within 28 working days of the submission deadline. The 28 calendar day deadline does not apply to work submitted late. Feedback can be accessed through the Turnitin assessment link on the course page. Further instructions on submitting an assessment and accessing feedback can be found on the University's VLE.

Late Submissions

Students are reminded to submit their assessment in the correct format and ahead of the published deadline. Deadlines are strict and Canvas uploads made remotely might not be immediate, we therefore strongly recommend that students upload their work to Canvas in good time before the deadline. If assessments are submitted late without approved Extenuating Circumstances, there are penalties:

- For assessments submitted up to two days late: any mark of 40% or higher will be capped at 40% for undergraduate students. Any mark of 50% or higher will be capped at 50% for postgraduate students. Any mark below 40% for undergraduate students and below 50% for postgraduate students, will stand.

- Students who do not submit their assessment within two days, and have no approved extenuating circumstances, are deemed not to have submitted and to have failed that assessment element. The mark recorded will be 0%.
- Late penalties are calculated differently for some types of portfolios. Please read the Assessment Brief of your portfolio carefully.

For further information, please refer to AQF7 Part C in the Academic Handbook.

Extenuating Circumstances

The University's Extenuating Circumstances (ECs) procedure is in place if there are genuine circumstances that may prevent a student submitting an assessment. If the EC application is successful, there will be no academic penalty for missing the published submission deadline.

Students are normally expected to apply for ECs in advance of the assessment deadline. Students may apply for consideration of ECs retrospectively if they can provide evidence that they could not have done so in advance of the deadline. All applications for ECs must be supported by independent evidence.

Students are reminded that the ECs procedure covers only short-term issues (within 21 days leading to the submission deadline) and that if they experience longer-term matters that impact on learning then they must contact Student Support and Development for advice.

Under the Extenuating Circumstances Policy, students may defer an assessed element on only one occasion and may request an extension on a maximum of two occasions.

For further information, please refer to the Extenuating Circumstances Policy in the Academic Handbook.

Academic Misconduct

Any submission must be a student's own work and, where facts or ideas have been used from other sources, these sources must be appropriately referenced. The Academic Misconduct Policy includes the definitions of all practices that will be deemed to constitute academic misconduct. This includes the use of artificial intelligence (AI) where not expressly permitted within the assessment brief, or in a manner other than specified. Students should check this policy before submitting their work. Students suspected of committing Academic Misconduct will face action under the Policy. Where students are found to have committed an offence they will be subject to sanction, which may include failing an assessment, failing a course or being dismissed from the University depending upon the severity of the offence committed. For further information, please refer to the Academic Misconduct Policy in the Academic Handbook.

Version History

Title: Assessment Brief Template **Approved by: The Quality Team**					
Version number	**Date approved**	**Date published**	**Owner**	**Location**	**Proposed next review date**
3.0	August 2022	August 2022	Registrar	VLE, Faculty Resources Page	July 2023
2.3	December 2021	December 2021	Registrar	VLE	August 2022
2.2	August 2021	August 2021	Registrar	VLE	August 2022
2.1	September 2020	September 2020	Registrar	VLE	August 2021
2.0	September 2020	September 2020	Registrar	VLE	August 2021
1.0	August 2019	August 2019	Registrar	VLE	August 2020
Referenced documents	AQF7 Academic Regulations for Taught Awards; Extenuating Circumstances Policy; Academic Misconduct Policy; Course Syllabus				
External Reference Point(s)	UK Quality Code Theme: Assessment				