National University of Singapore
School of Computing

IS3106 – Enterprise Systems Interface Design and Development
Assignment 1: Individual Assignment
Total Marks: 20 marks

LEARNING OBJECTIVES
At the end of this assignment, you should be able to:
- Perform backend development using Java EE technologies.
- Use JSF and Prime Faces to create a rich frontend presentation.

SYSTEM DESCRIPTION
In this assignment, you will be building a Library Management System (LMS). As this assignment is only worth 20% of the course grade, the set of features will be more limited compared to what we see on most library management systems. This assignment is mainly used to assess your ability to perform both backend and frontend development using the Java technologies that were covered in Labs 1-4.

The list of functional requirements is shown below, and you will be mainly assessed based on the required functional requirements.

GENERAL REQUIREMENTS
- This is an individual assignment. You are supposed to work on it on your own. Please refrain from discussing with your friends. *Your submission will be assessed carefully to ensure that there is no plagiarism (both with your classmate's submissions and sample codes online).*
- You must develop using **Java EE** and **Prime Faces** in the NetBeans environment. You are required to submit your NetBeans project.
- You are free to use any images and templates [1] that you find online. Your assignment submission will not be published online so you do not need to be concerned with copyrights. However, if you choose to deploy the project for your own use later, you have to ensure that the media files do not infringe copyright. We will not be liable for any copyright infringement.

BACKGROUND
The West Coast community centre is planning to establish a community library to improve community engagement by providing free access to books and study areas. In order to streamline the lending process of books and to reduce the workload of the staff, the community management has decided to incorporate a new LMS. The LMS will consist of i) a core backend to be developed with a component-based architecture, and ii) a rich frontend to be developed with JSF and Prime Faces iii) a book lending application to support the workflow of the library. During the initial phase of this project, you are tasked to develop a pilot system at the library to demonstrate the solution. In a typical scenario, when a new member walks into the library, he/she will be assisted by an admin staff at the lending counter. Initially, the staff will register him/her with the system using his/her national identity number (NRIC/FIN) upon which the member can lend books.

[1] Use of templates and themes based on CSS/JavaScript is fine but you should refrain from copying large chunks of codes from online or sample projects that you might obtain from online.

SYSTEM SPECIFICATION

The pilot system functionalities to support the use cases depicted in the UML use case diagram as shown in Figure 1.

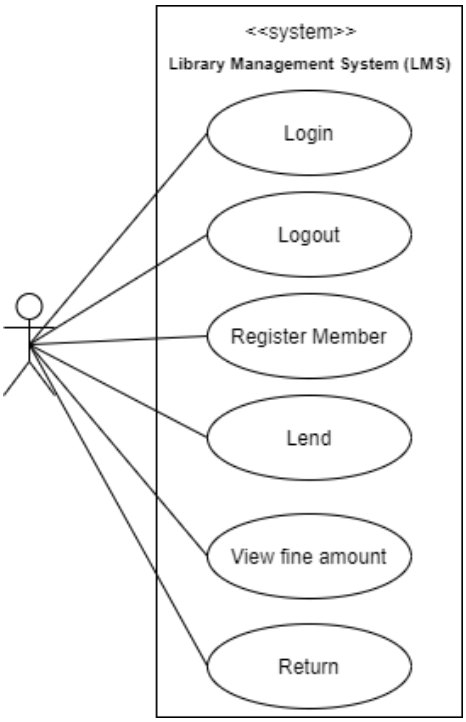

Figure 1 – Use Case Diagram

A brief description of each business use case is given in Table 1 below.

S/N	Use Case	Description/Business Rules
1	Login	• May only be performed if the staff is not currently logged into the system. • Staff must be currently logged into the system to perform all other use cases.
2	Register Member	• If a member is not registered, the admin staff registers a new member record in the system. The staff needs to ask and enter the following information: ○ First Name ○ Last Name ○ Gender ○ Age ○ Identity Number (NRIC or Passport) ○ Phone ○ Address
3	Lend Book	• Admin staff can lend books available in the library, to the members.
4	View fine amount	• If the return date is later than two weeks, calculate the fine and show the fine amount for the lending.
5	Return Book	• When a member returns a book, store the book return date and the fine amount.
6	Logout	• Logout if the staff is currently logged in the system.

Table 1 – Use case descriptions and business rules of ILS

The logical data model of the LMS is depicted in the UML class diagram shown in Figure 2. The entity classes are required to implement system functionalities for supporting the business use cases.

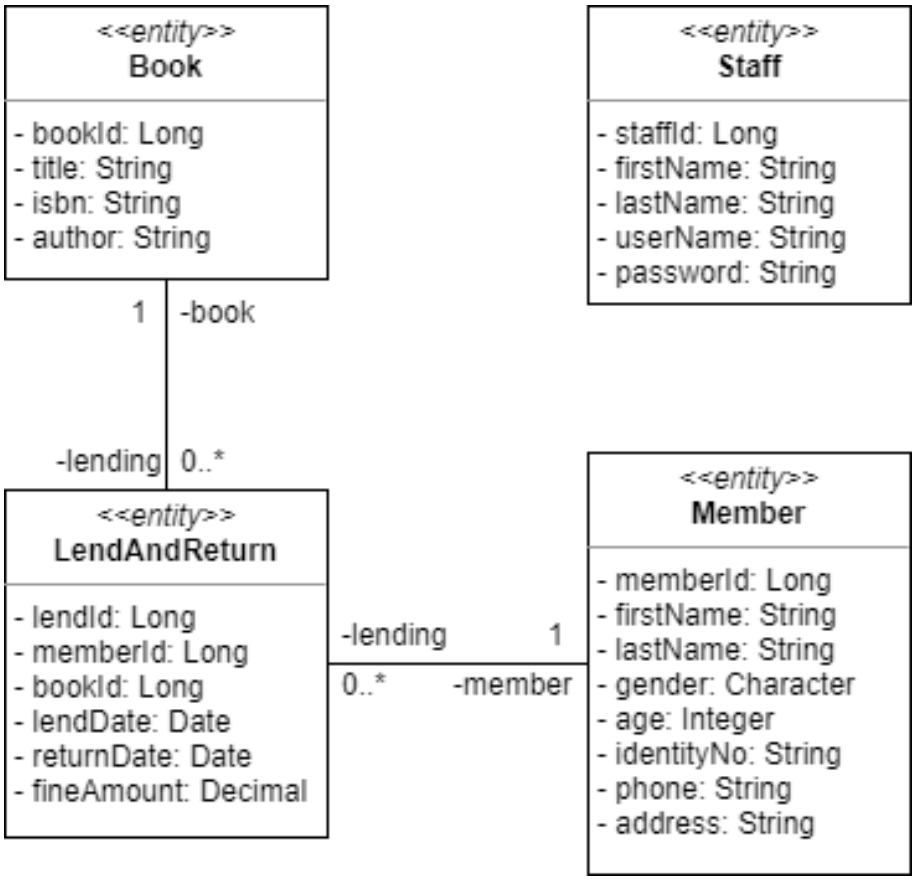

Figure 2 – Class Diagram

Data init

For the pilot implementation, the library has asked you to deploy the solution at the library. There are several employees as shown in Table 2.

staffId	firstName	lastName	userName	password
1	Eric	Some	eric	password
2	Sarah	Brightman	sarah	password

Table 2 – Staff Entities

A sample of the books available at the library is given in Table 3.

bookId	title	isbn	author
1	Anna Karenina	0451528611	Leo Tolstoy
2	Madame Bovary	979-8649042031	Gustave Flaubert
3	Hamlet	1980625026	William Shakespeare
4	The Hobbit	9780007458424	J R R Tolkien
5	Great Expectations	1521853592	Charles Dickens
6	Pride and Prejudice	979-8653642272	Jane Austen
7	Wuthering Heights	3961300224	Emily Brontë

Table 3 – Book Entities

In addition, the library has also provided you with sample data of their member database as shown in Table 4.

memberId	firstName	lastName	gender	age	identityNumber	phone	address
1	Tony	Shade	M	31	S8900678A	83722773	13 Jurong East, Ave 3
2	Dewi	Tan	F	35	S8581028X	94602711	15 Computing Dr

Table 4 – Member Entities

YOUR TASK

The above is the list of functions that your system should support. The assessment will be mainly based on these functionalities. You are allowed to include additional/bonus features. For simplicity, consider the following assumptions:

- If a book is already lent to a member, it cannot be lent again.
- The lending period is two weeks (14 days).
- For the delay of returning, per day 50 cents will be fined.
- Members can return a book even within the lending period.
- When lending a book, the return date and fine amount in the LendAndReturn entity should be NULL and 0 respectively.
- When returning a book, a member must pay the available total fine amount. If not book cannot be successfully returned.
- Consider following exception handling classes in your application
    - EntityManagerException.java
    - InvalidLoginException.java
    - StaffNotFoundException.java
    - BookNotFoundException.java
    - MemberNotFoundException.java
    - LendingNotFoundException.java
    - FineNotPaidException.java

IMPLEMENTATION CONSIDERATIONS

1. You must use MySQL as the relational database to implement data persistence. **You MUST set up the database connection with the following properties for proper testing of your project. Please make sure the sample data is hardcoded in your program to be programmatically inserted into the database during the initial deployment of the project.**
    a. Database Name: **drrsdbs**
    b. Database user: **root**
    c. Database password: **password**

2. The date format is yyyy-MM-dd (i.e. 2019-02-21) and the time format is HH:mm (i.e. 12:30)

3. The list of features shown above is not exhaustive. You can also consider adding additional features or enhancements to the above. These additional features/enhancements will count towards the bonus features of up to 5% of the course grade.

SUBMISSION

You are required to upload your complete NetBeans project folder (as a zip file) to Canvas Workbin Assignment 1 Submission folder by **17 March 2023 23:59** hrs (The grace period of 15 minutes). Failing to do so would render you zero marks for the assignment. Name your file with your Matriculation number. Include readme.txt describing any extra features you have incorporated into your system.

EVALUATION

The following grading criteria are used for the evaluation of the assignment:
Weightage: 20% of the course grade.

- 13% Features
    - Different weightage will be given for different functions depending on their difficulty
    - Grading is mainly based on correctness rather than code quality
- 7% - UI/UX and aesthetic
- Up to 5% for any bonus features (Top-up to a max of 20%)