Ton Duc Thang University

Enterprise Systems Development Concepts

LIBRARY MANAGEMENT SYSTEM

Lecturer:
DUONG HUU PHUC
Students
Ngo Quoc Trung
Nguyen Dinh Than

December 3, 2017

1 Description

The main entities of the library management system are book, invoice, employee, student. The library management system has books, users (employee and student), invoices. Attributes of the book are id, name, available, numbers. Attributes of the employee are name, id, password, role. Attributes of the invoice are id, id_book, id_student, id_employee, numbers, begin, end. Attributes of the student are id, name. A student can order one or many invoices. Invoices can be ordered by a student. A book can be included by one or many invoices. The invoice contains only a book. The invoice only has an employee. The employee can have one or many invoices.

The library management system provides real-time information about the availability of books in the library. The library management system is run on the console. When the employee (admin, staff) want to start doing something, the user will need to sign in the system.

The student can borrow only a book per time, for the period of one week. The student can borrow a limited number of books (same names) is one. A book can be borrowed by many users. The student must return book(s) which they borrowed. The student will need their Library ID card to get items from the library. Unless the students have ID card, they can come to library desk and ask for an ID card. The student can search the book they want to borrow. If a student wants to borrow a book. First, they come to the library and ask staff to borrow a book. The staff will sign in to a system and then select a function "Allow user to borrow a book" and they write the name of the book, student's name. Infomation about borrower, staff, and book will be saved in the database. If a student wants to return a book, he/she will come to the staff again. The staff will check borrower's name and book's name. If everything is correct, the staff will allow customer to return a book. Books that are returned late will begin to accrue overdue fees. Books are managed by admin. The administrator of library can update, delete, add books. Not only the admin can reach all borrowers but also can add a new employee to the system.

2 Specification

Use case: Sign in
Description: Use case allows user sign-in system.
Actor: staff,admin
Request: Login name and password must match the registration information
Preconditions: The actor has an account.
Postconditions: If the use case was successful, the actor could log in system
Basic Flow :
- The system show command line for the actor enters name and password.
- The actor will enter name and password
- The system validates the entered name and password.Then the actor to go into the system.
Alternative Flows :
If the user's name or password isn't correct, the system will print "Wrong username /password !" and exit.

Use case: Book Title Lists
Description: Use case allow actor can see all books in library
Actor: Student.
Request: User selects a function "Book Title Lists".
Preconditions: User selects a function "Book Title Lists".
Postconditions: The actor can see all books in the library.
Base Flow :
- The system shows all books with status in the library.
Alternative Flows :
Nothing

Use case: Change Password
Description: Use case allows actor can change the password.
Actor: Staff, Admin
Request: The user has an account.
Preconditions: The actor complete use case "Sign in".
Postconditions: The actor can change the password.
Base Flow :
- The system show command line for the actor enters a new password.
- The actor will enter a new password
- The password is changed.

Alternative Flows :
If the system has a something error, the system will print “Something wrong.Please try again”

Use case: Search a book
Name: Search a book
Description: Use case allows actor can search a book by name
Actor: Student
Request: User selects a function “Search a book”.
Preconditions: User selects a function “Search a book”.
Postconditions: The actor searches a book.
Base Flow :
- The system show command line for the actor enters the name of the book.
- The actor will enter the name of the book.
- The system will show all books which match with a name.
Alternative Flows :
Nothing

Use case: Add an employee
Name: Add an employee
Description: Use case allows actor add a new employee.
Actor: admin
Request: The actor has an account – admin account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor adds a new employee
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the employee, the password of the employee, the role of the employee.
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If the name of the user already exists in the system
- The system print “Something wrong.Please try again”.
Use case: Add books
Description: Use case allows actor add books.
Actor: admin
Request: The actor has an account – admin account.
Preconditions: The actor complete use case “Sign in”.

Postconditions: The actor adds books.
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the book, the status of the book, a number of the book.
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If The name of your book already exists in the system :
- The system print “Something wrong.Please try again”.

Use case: Delete books
Description: Use case allows the actor delete books from the library.
Actor: admin
Request: The actor has an account – admin account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor deletes books from the library.
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the book.
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If the system has a something error, the system will print “Something wrong.Please try again”.

Use case: update books
Description: Use case allow actor update books.
Actor: admin
Request: The actor has an account – admin account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor update books.
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the book, the status of the user, a number of the user.
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If the book’s name isn’t correct, the system will print “Something wrong.Please try again”.

Use case: Get All Borrowers
Description: Use case allow actor can see all borrowers in library
Actor: admin
Request: The user has an account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor can see all borrowers in the library.
Base Flow :
- The system shows all borrowers in the library.
Alternative Flows :
Nothing .

Use case: Allow user to return a book
Description: the actor allow the user to return a book.
Actor: staff
Request: The user has an account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor allows the user to return a book in the library.
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the book, the name of the borrower.
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If the book’s name isn’t correct or the user’s name isn’t correct, the system will print “Something wrong.Please try again”.

Use case: Allow user to borrow a book
Description: the actor allow the user to borrow a book.
Actor: staff
Request: The user has an account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor allows the user to borrow a book in the library.
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the book, the name of the borrower.
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If the book’s name isn’t correct or the user’s name isn’t correct, the system

will print “Something wrong.Please try again”.

Use case: Add a student
Name: Add a student
Description: Use case allows actor add a new employee.
Actor: staff
Request: The actor has an account – staff account.
Preconditions: The actor complete use case “Sign in”.
Postconditions: The actor adds a new student
Base Flow :
- The system show command line for the actor enters.
- The actor will enter the name of the student
- The system validates the entered data.The system print “Done!”.
Alternative Flows :
If the name of the user already exists in the system :
- The system print “Something wrong.Please try again”.

3 Diagram

3.1 Use Case

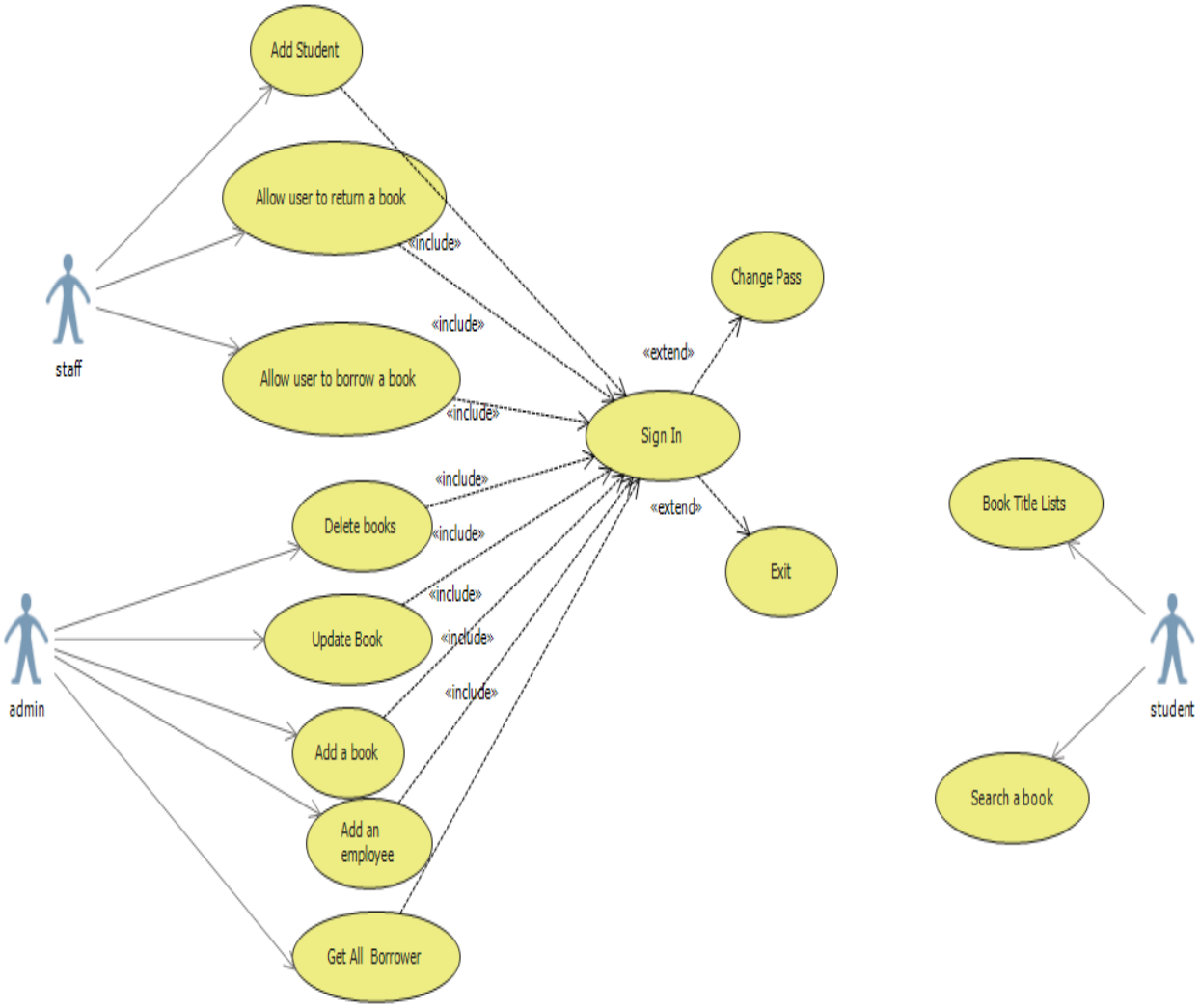

Figure 1: Use Case Diagram

3.2 ERD

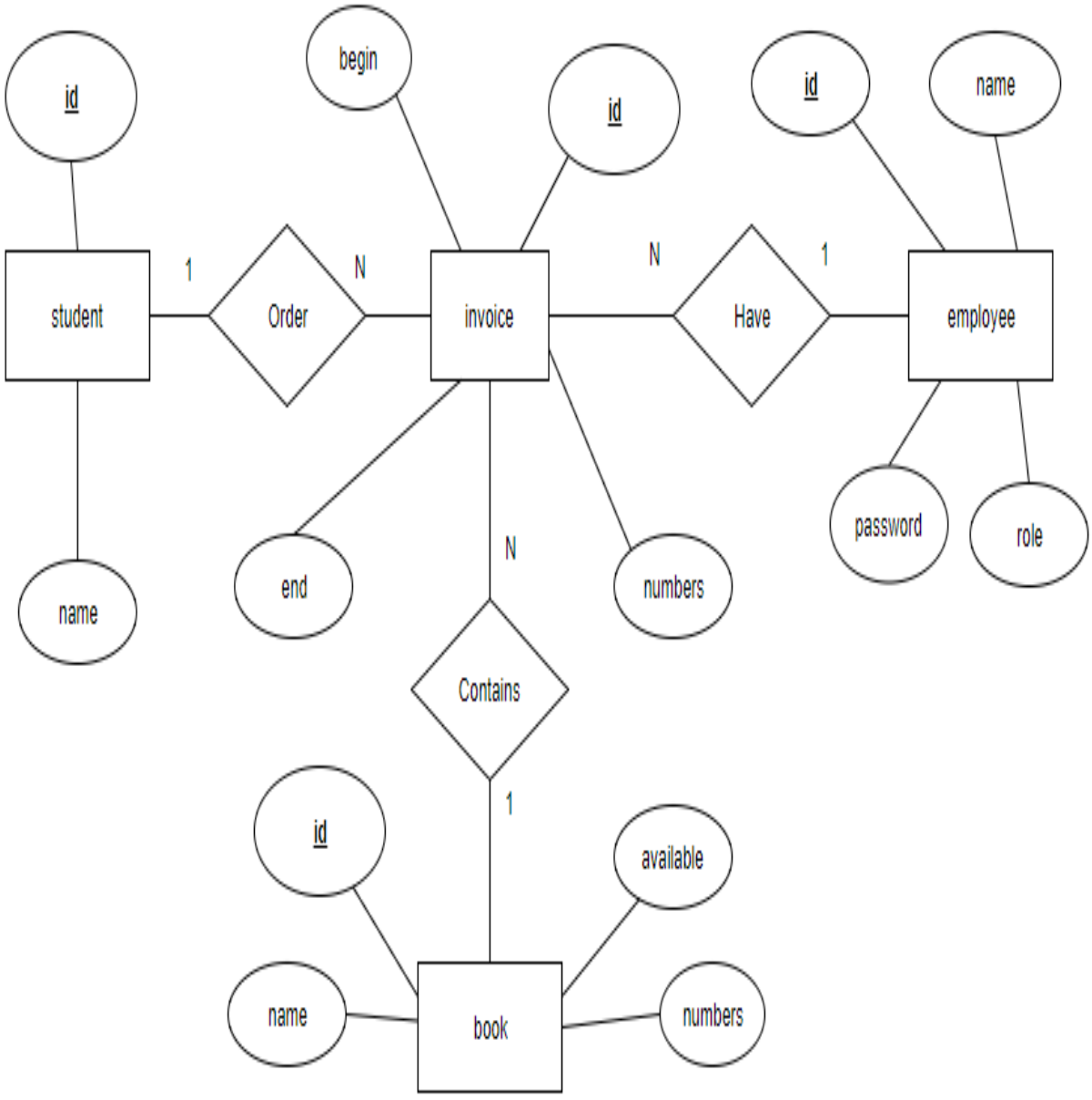

Figure 2: ERD Diagram

3.3 Data

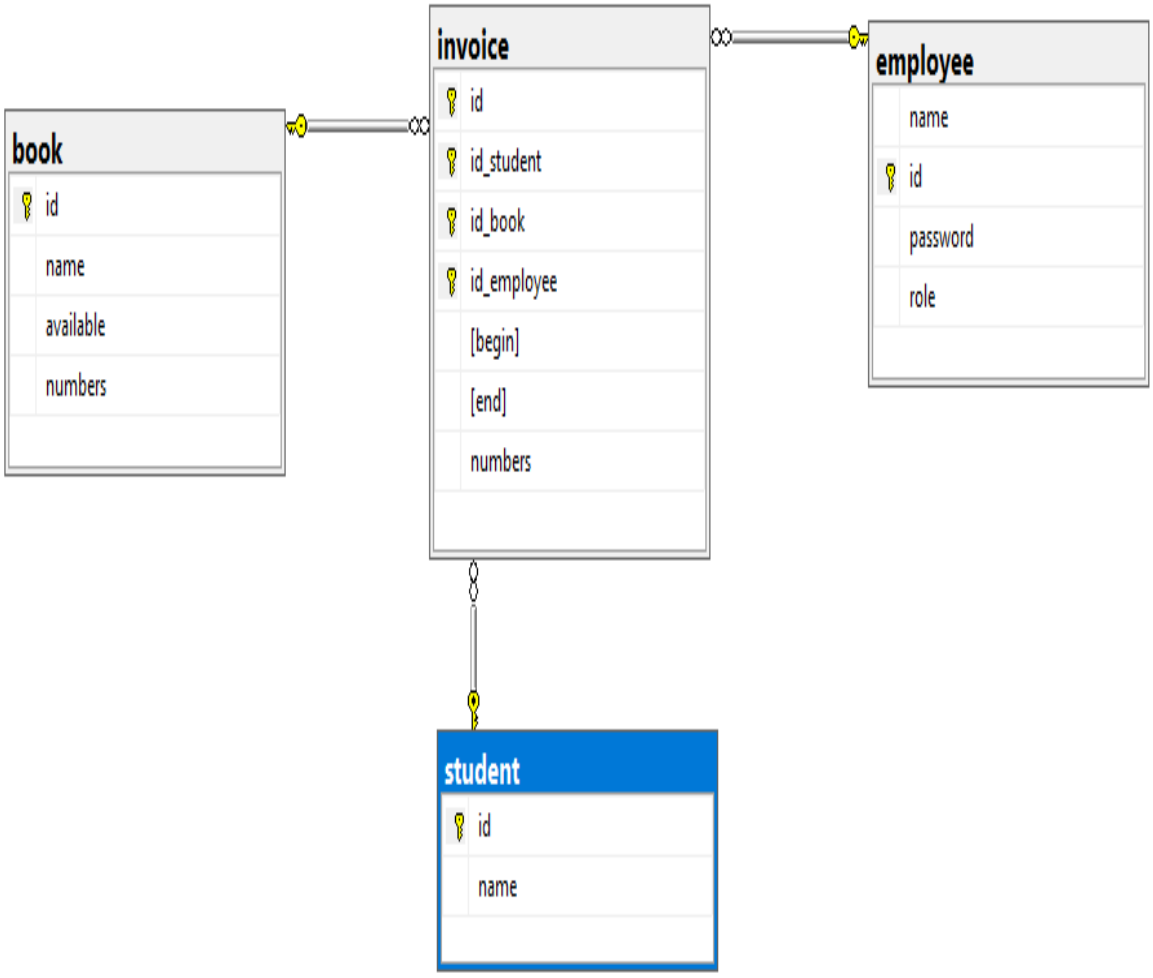

Figure 3: Data Diagram