Software Design Specification of Library Management System for OOD

Prepared by Li Zhenbang

Waseda University

2017.12.6

Table of Contents

1. Introduction

This library management system is designed in Java and use a JDBC to connect to the MySQL. This is an assignment for OOD. In the following document, I will show the designing details in the designing UML or the ordinary thought in my brain.

1.1. Document Outline

Here is the outline of the library management system. In this document, I will introduce each section of the system and I will give a overall structure to shoe what the system is like, also with some UML diagrams to descript it. In the end of the document, I will solve some problems and clarify them in an OOD aspect.

1.2. Document Description

1.2.1. Introduction

The library management system is developed for the day-to-day operation of the school's library management software. According to the needs of administrators and readers borrowing, the orderly and informative management of books and readers is realized. A module consists of: daily borrowing management, reader management, library management, system settings. Library management system by recording the details of the book, the establishment of reader files, system user information, set various parameters (such as: deadline, type of personnel, book categories, etc.), identify the correspondence between books and readers to achieve daily Lending operation. Based on these basic data obtained, the system generates the corresponding statistical data for administrators to query and analyze. In addition, the administrator can also carry out regular data updates and database maintenance of these basic information; to provide readers with the library is borrowed books, as well as borrow their own books, books and other inquiry functions, library management system to the librarian Provides a simple, quick and easy way to manage these huge, cumbersome information and books.

2. Design Considerations

2.1. Assumptions and Dependencies

UML tool: Astah Community
Structure diagram and Sequence diagram tool: ProcessOn
Login UI: Adobe PS
Programming Environment: Java JDK1.8, Eclipse EE
Database: MySQL
Operate System: winxp or more

2.2. General Constraints

The name of the software is with beginning as “LM_” and the database will be unified in the future.

2.3. Goals and Guidelines

Library management system business processes mainly for system management processes, book business processes and reader business processes.
After entering the user name and password, the system administrator logs in the system and the main interface of the system appears. The submenu in the menu bar includes all the functions of the system.
After selecting the system management menu, the administrator can write to the administrator user
Interest to add, delete and modify password management operations.

After selecting the library management submenu, the administrator can perform various operations on the management of book information. For example, new books may be stored in the library after the new books have been added. When borrowing books, readers borrow books. Readers also return the administrator to the book operation. Reader renewals by the administrator to renew the operation. When the book information needs to be modified or deleted, the administrator will modify or delete the book information. With the Reader Management submenu selected, the administrator can perform various operations on the reader's information management. For example, with new readers, administrators can add new readers.

2.4. Proposed System

Proposed system is an automated library management system. Through our software user can search authors, search books, update information, edit information, borrow and return books in quick time. Our proposed system has the following advantages:

>User friendly interface
>Fast access to database
>Less error
>More Storage capacity
>Search facility
>Look and feel environment
>Quick transaction

All the manual difficulties in managing the library have been rectified by implementing computerization.

2.5. Development Methods

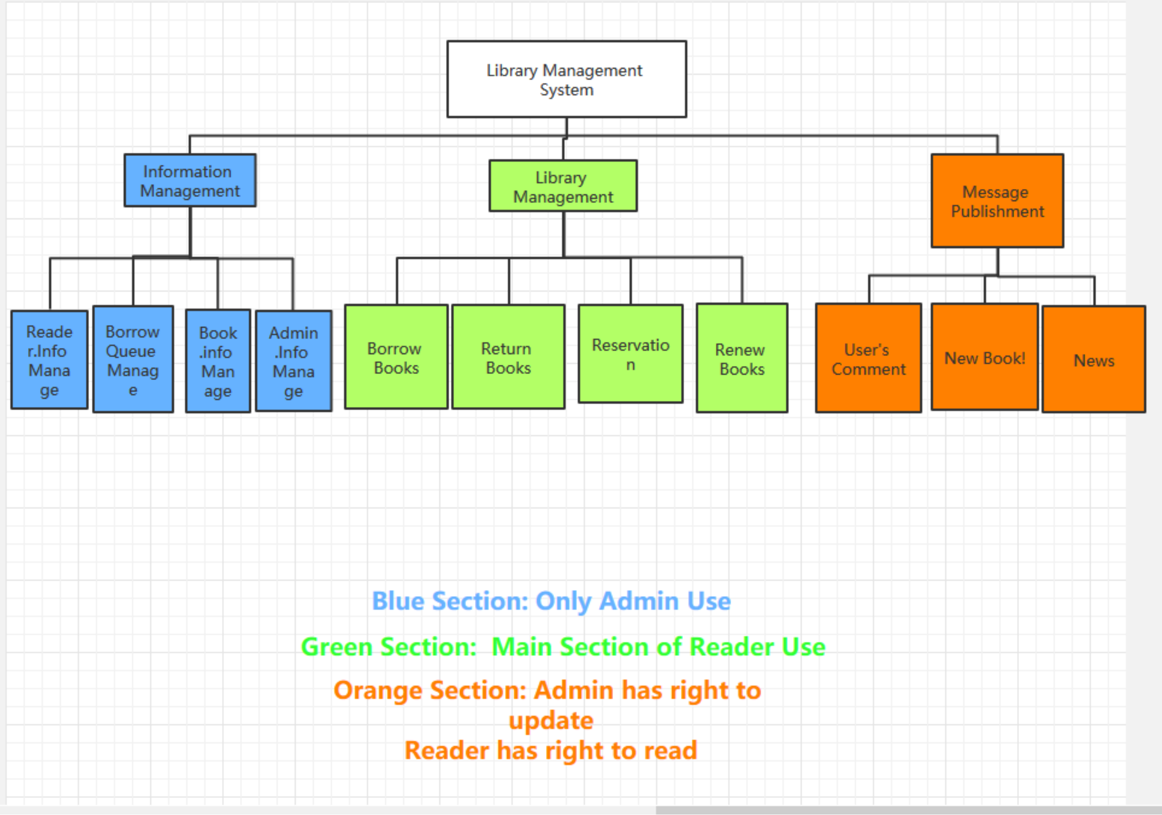
The method or approach used for this software design is the J2EE model. MVC architecture is used in this system and use JDBC to connect to the MySQL database. If several methods were seriously considered, then each such method should be mentioned, along with a brief explanation of why all or part of it was used or not used.

3. Architectural Strategies

3.1 Overall of document

The overall system design objective is to provide an efficient, modular design that will reduce the system's complexity, facilitate change, and result in an easy implementation. This will be accomplished by designing a strongly cohesion system with minimal coupling. In addition, this document will provide interface design models that are consistent, user friendly, and will provide straight forward transitions through the various system function

3.2 Overall Structure Diagram



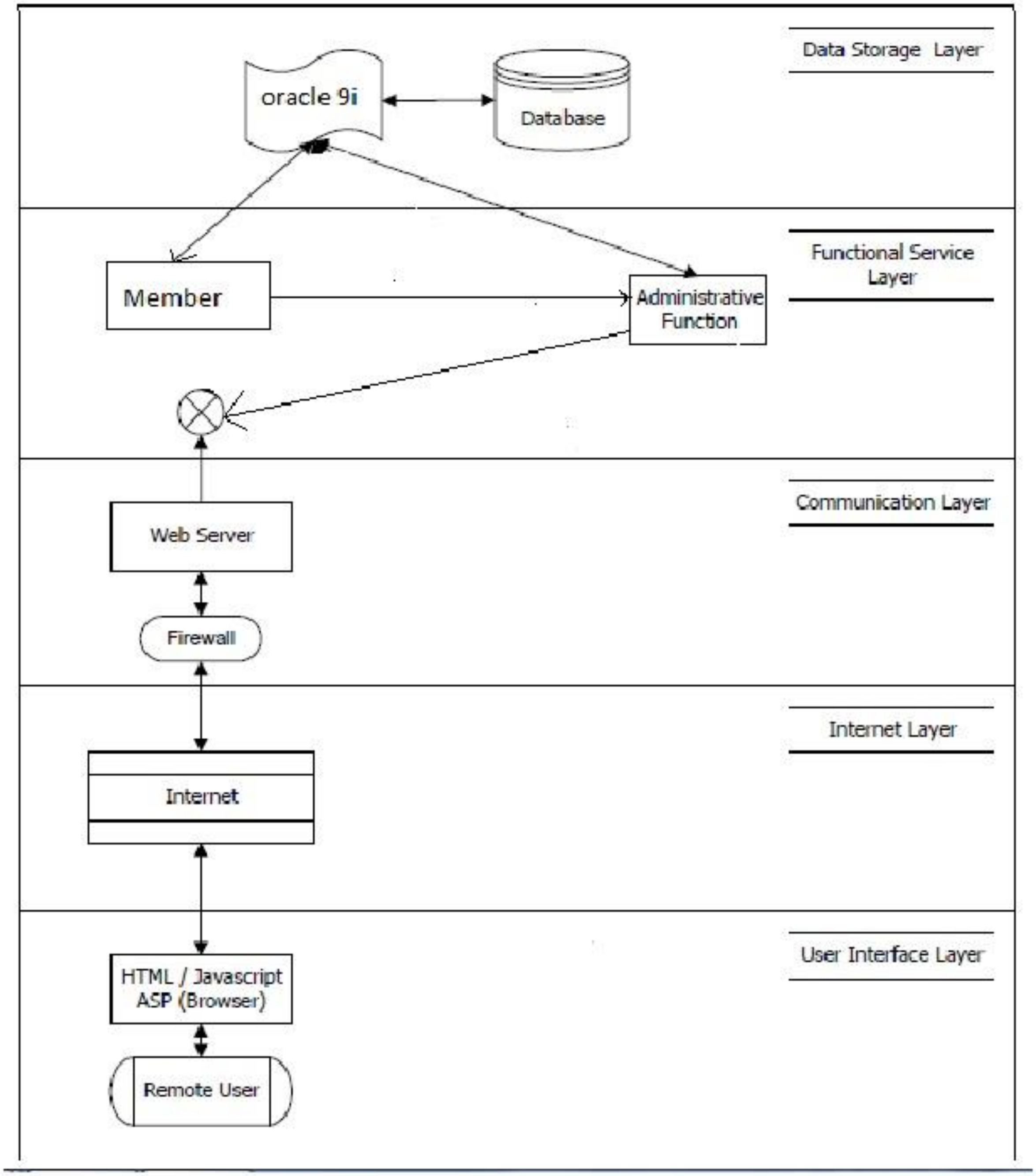
Blue Section: Only Admin Use

Green Section: Main Section of Reader Use

Orange Section: Admin has right to update
Reader has right to read

I am thinking about the interface of the message publishment, so, in the following diagrams, the interface of the message publishment is omitted.

3.3 System Architecture

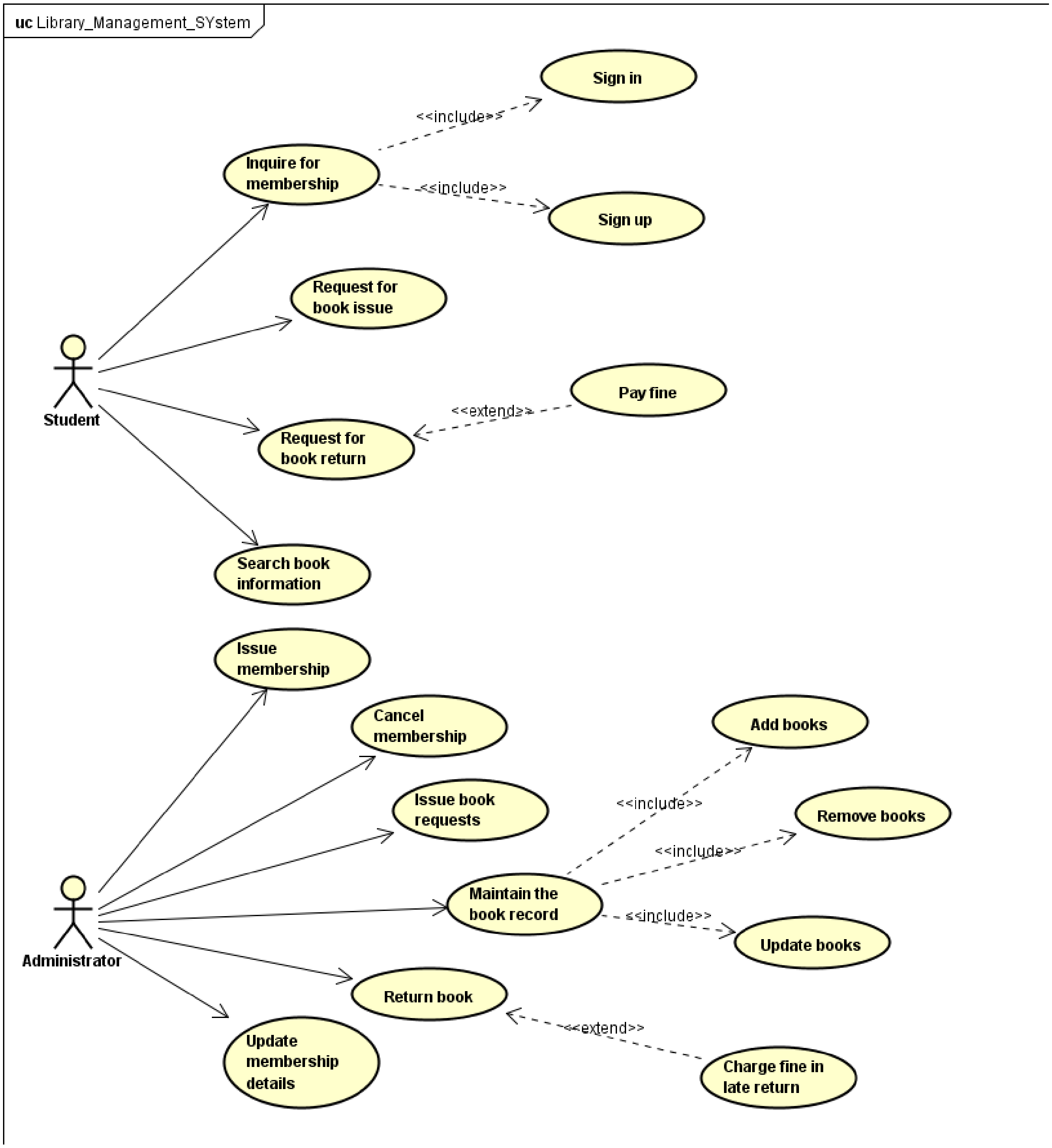


The online Library System is a client-server based system, which contains the following layers: user interface, internet communication, functional service, and data storage layers.

Data transfers occur in both directions in the system. The users input or data request is sent using either an internet browser or through the windows client. This data then connects to the system either through the internet or, in the case of an onsite connection, through the LAN connection. In the case of an internet connection, the data is required to pass through the system's firewall, for security purposes, prior to connecting to the web server. Local personnel, once validated within the system, will be connected directly to the application server. In the functional services layer, the data input or request is routed to the appropriate functional module in accordance with the user login and account type. Through these modules, the users will interact with the database via MySQL.

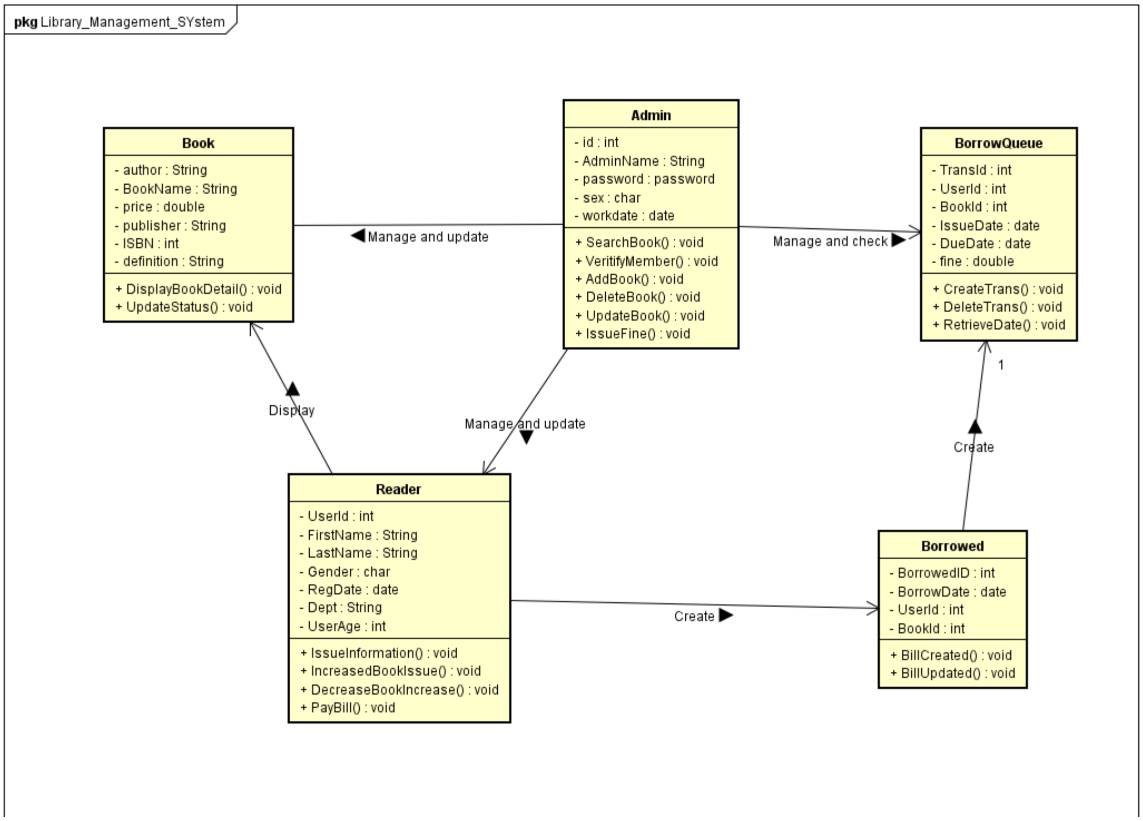
4.System Designing Diagram

4.1 User case diagram

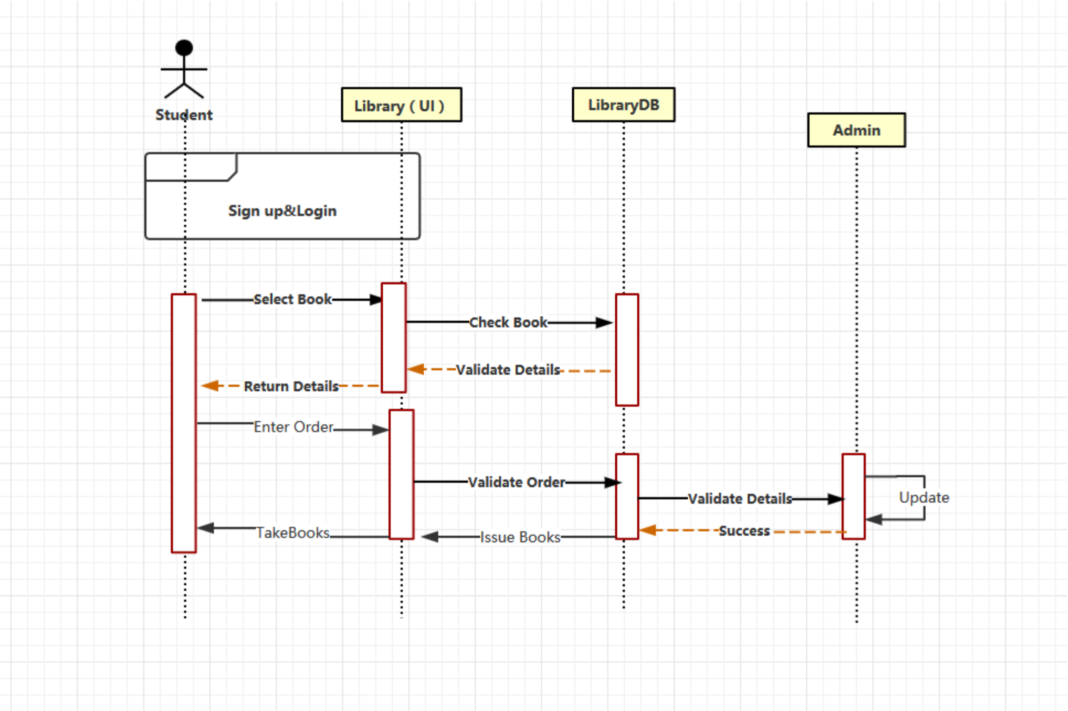


In this diagram, I chose 2 roles and showed what they can do in the system. As I have showed before, the Message Publish part is omitted temporarily.

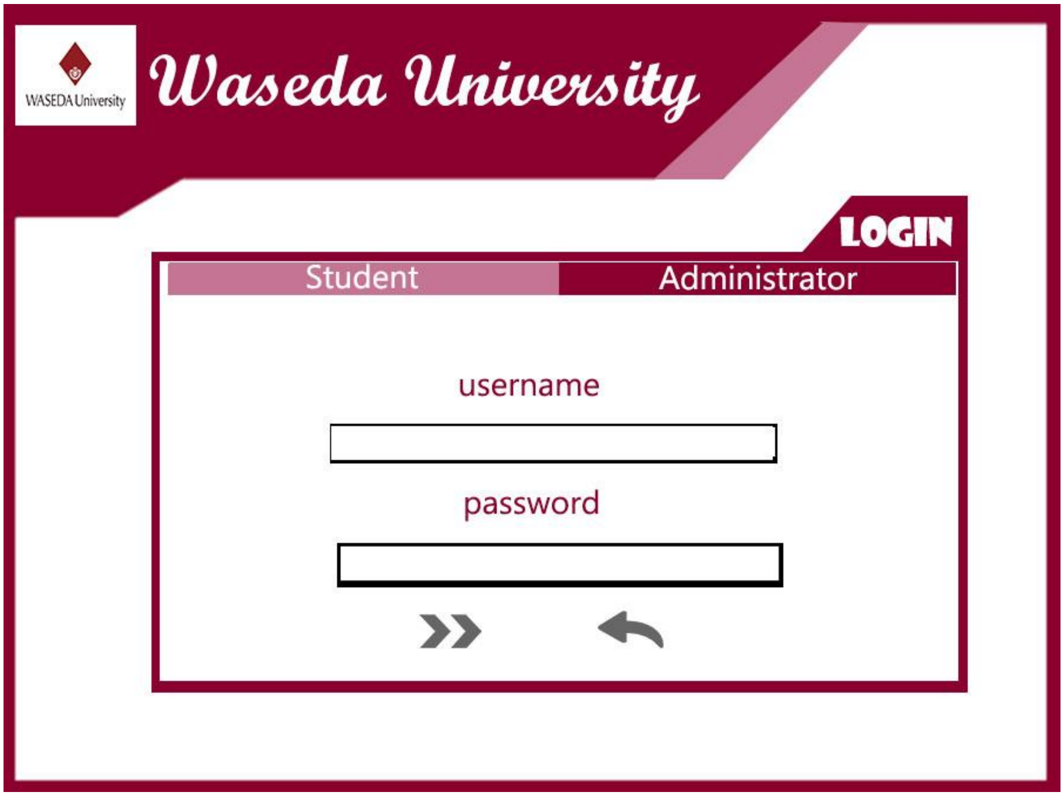
4.2 Class diagram



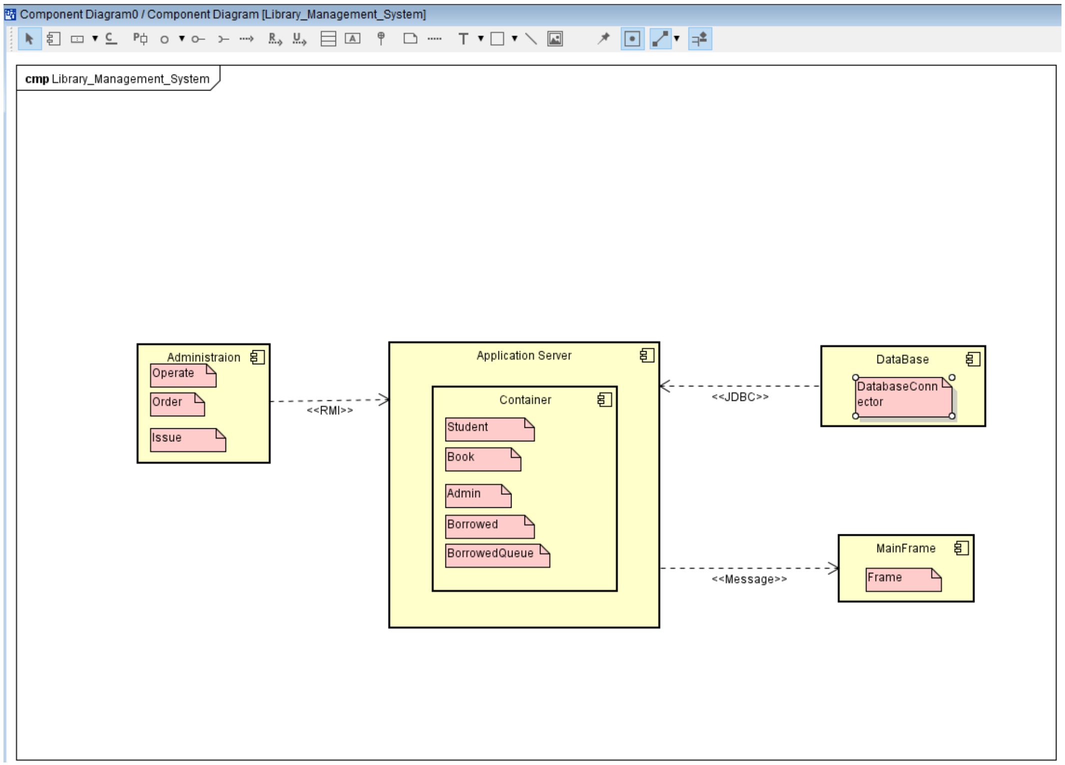
4.3 Sequence diagram



4.4 Login UI concept diagram

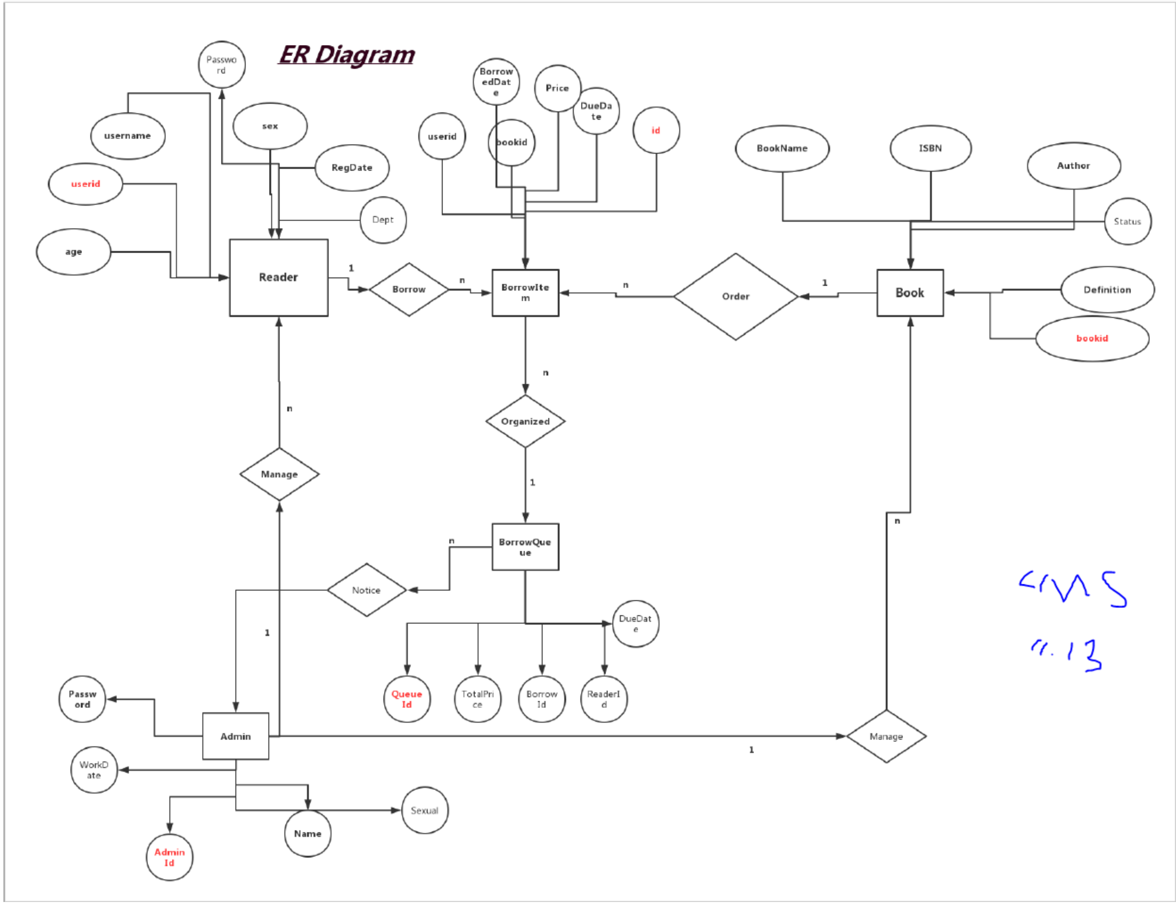


4.5 Component Object Model



4.6 DataBase Designing

ER diagram



cms

4.13

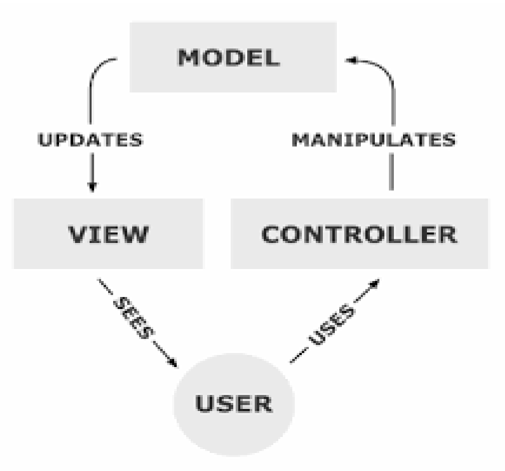
Red words are main keys and the foreign keys are not noted in the diagram.

5 Development model design

5.1 J2EE

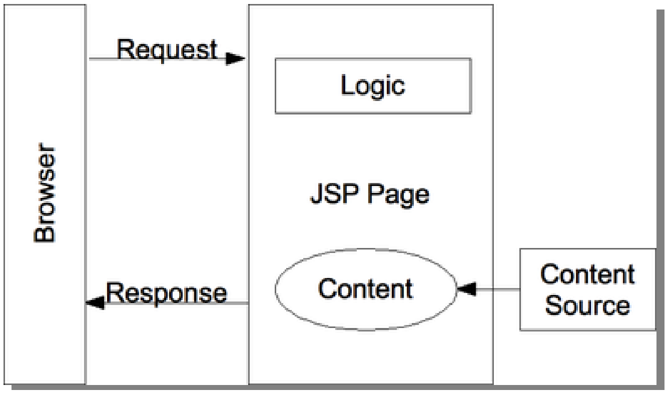
J2EE is currently one of the major platforms for developing web applications. From the overall structure point of view, J2EE platform uses the MVC model. MVC model the whole system is divided into three layers: model layer, view layer, controller layer. The model layer represents the enterprise data and business logic. The view layer is the interface between the user and the system. The controller layer plays a role of communication between the model layer and the view layer, handles the user input on the view layer and controls the view layer And model layer data stream and the view layer page flow.

The benefit of this model is that by separating the data and its representation, separating the control logic from the presentation interface, the looseness between the various modules in the system is increased, increasing the reusability and maintainability of the code.

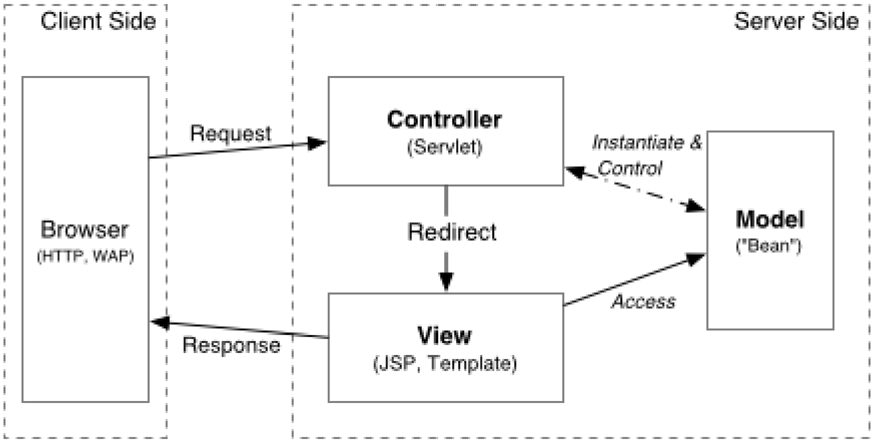


5.2 JSPModel

JSP specification JSPModel1 and JSPmodel2 proposed two ways to build applications, their essential difference is that the location of processing different batch requests. In the model1 system, the JSP page solves the request and returns the result to the client. All data access is done by JavaBeans.



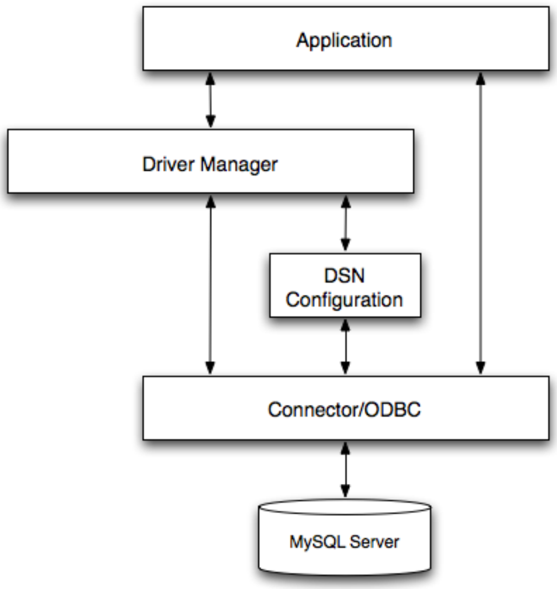
The Model 2 architecture, shown here, is a way to combine JSP and servlets to implement dynamic content services. It clearly separates the expression and the content, defines the definition of roles and the division of labor between developers and web designers. The model1 system is very suitable for simple application needs but cannot meet the complex application of large-scale implementation



The system uses model1 method, but in the JSP page, the introduction of servlet.jsp page to simulate model2 has the scalability and strong adaptability.

5.3 DAO Architecture

DAO component's main function is to provide data storage services, he completed the connection to the database configuration and JDBC package



Due to job time constraints, the system does not have a real servlet to control page scheduling, data access, etc., but by a servlet.jsp page to achieve

5.4 The architecture of system

First of all, servlet.jsp from the page using the jsp object to get the last page passed parameters. Then the corresponding field name of the extracted string type parameter is put into the hash table in the form of key-map type. The key-map pair is the data structure of the hash table, and then the hash table is used as the parameter to call the execution provided in the bean correspondingly Process function of the way the implementation of the operation. This servlet.jsp achieved its role as a controller.

6 Feasibility Analysis

Hardware configuration (At least)
Processor: Pentium III 630MHz
RAM: 128 MB

Hard Disk: 20GB
Monitor: 15" Color monitor
Key Board: 122 Keys

Software Configuration (At least)
Operating System: Windows NT/Win 98/ Win XP or more
Language: Java 2 Runtime Environment
Database: MySQL server 5.7

7 Glossary

JDBC: The Java Database Connectivity The Java Database Connectivity API is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases

MVC：The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller.

J2EE: J2EE is a platform-independent, Java-centric environment from Sun for developing, building and deploying Web-based enterprise applications online.

LMS – Library management system
HDD - Hard Disc Drive
RAM – Random Access Memory
IE – Microsoft Internet Explorer
SQL – Structured Query Language
DD – Design Documentation

8 Bibliography

[1] Object-Oriented-Analysis-and-Design-with-Applications-3rd-Edition
[2] IEEE software designing document template
[3] Information architecture of JAVA, China Posts and Telecommunications Press 3rd edition