Hacettepe University
Department of Computer Engineering
BBM478 - Software Engineering Laboratory

Software Design Document

Group One

Ahmed Şamil BÜLBÜL (21426749, Software Developer)
Halil İbrahim ŞENER (21328447, Tester)
Naciye GÜZEL (21580841, Project Manager)

1.   REVISION HISTORY

Version	Date	Author	Change Description
1.0	25.04.2017	Ahmed Şamil Bülbül	Sequence Diagrams added. Annexes created.

2. INTRODUCTION

2.1 Purpose and Scope

This document will explain details of software design. The purpose is to give general information about the design of software and walk through main details of design such as interfaces.

In this document, topics that are covered are; how to make design constraints and decisions, which user interfaces are going to be implemented, how user interfaces will interact with each other, what are the general software components, how will software behave, what are the main data models and how will they behave.

2.2 Document Overview

Design constraints and decisions will describe the general constraints implied by the design process and what are the impacts on the system architecture and modules design. Hardware limitations and software environment constraints will be described here.

Design details will explain software's general architectural design with data models, ER diagram, user interfaces and their explanation, software components, and its behavior.

2.3 System Overview

2.4 Definitions, Acronyms, and Abbreviations
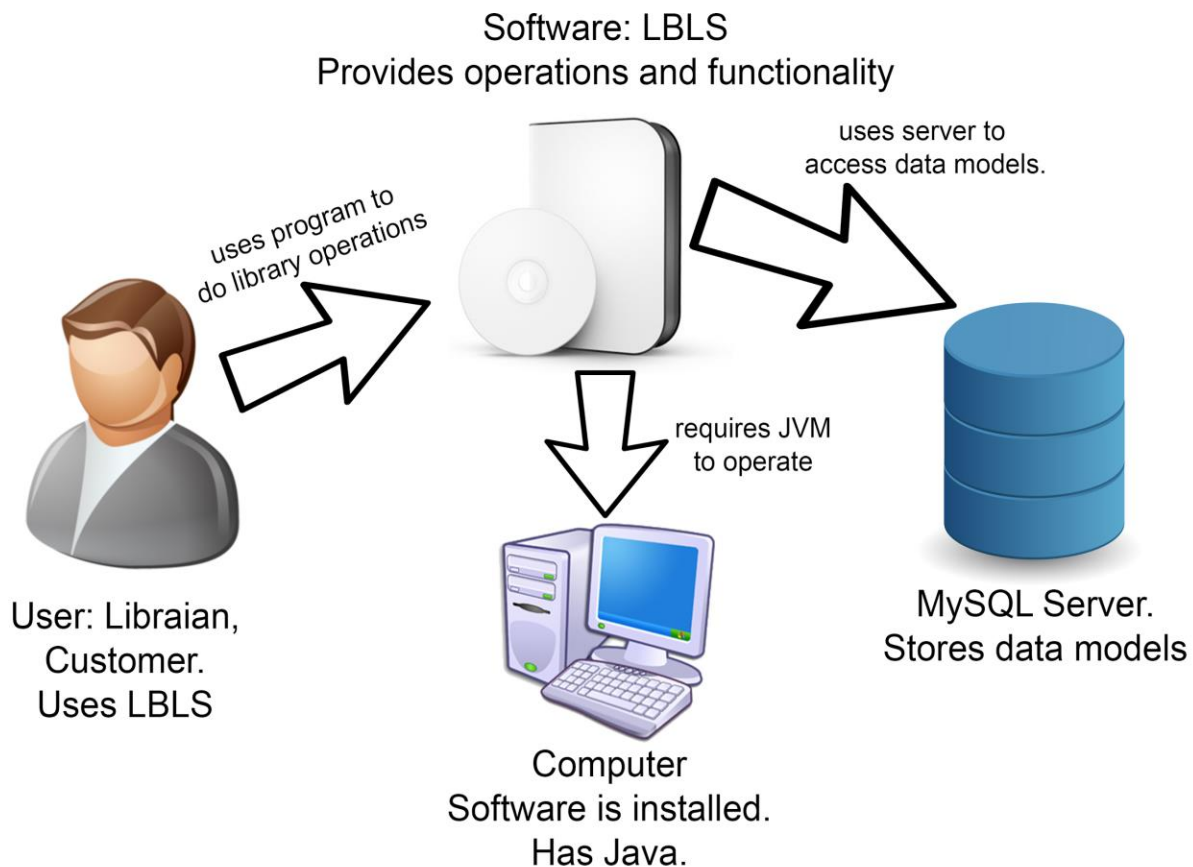
Term/Acronym	Definition
SRS	System Requirements Specifications
SDD	Software Design Description
SQL	Structured Query Language
JVM	Java Virtual Machine
LBLS	Library Book Loan System

3. Design Constraints and Decisions

- The system has to be develop using Java programming language.
- LBLS will be able to run in every operating system that runs Java.
- MySQL database will be used in the system.
- The LBLS will be a stand-alone desktop application that requires a database connection.
- Model-View-Controller (MVC) will be used as an architectural pattern.
- The system will be distributed over GitHub, documentations will also be on GitHub.
- The system will provide functionality and easy-to-use graphical user interface.
- Manipulating users can only be done by librarians.
- Manipulating book can only be done by librarians.
- Guest users can only use one functionality which is "Search for Books".
- The performance of the system should not decrease under multiple users logged in.
- Security of user logins will be provided by SHA-256. (Secure Hash Algorithm 2)
- The application will be protected from SQL injection attack by utilizing Java's PreparedStatement class.
- Development will be made in Eclipse integrated development environment. Eclipse provides graphical user interface development tools. GUI will be prepared by this.
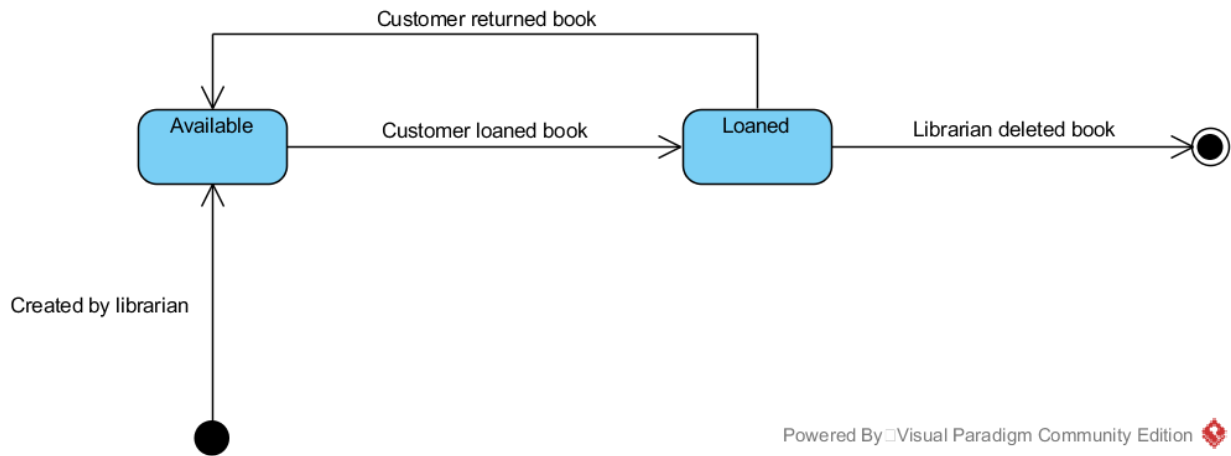
4.   Design Details

4.1   Software Components

Please refer to "SDD - Class Diagram" to see class diagram.

4.2   Software Behavior

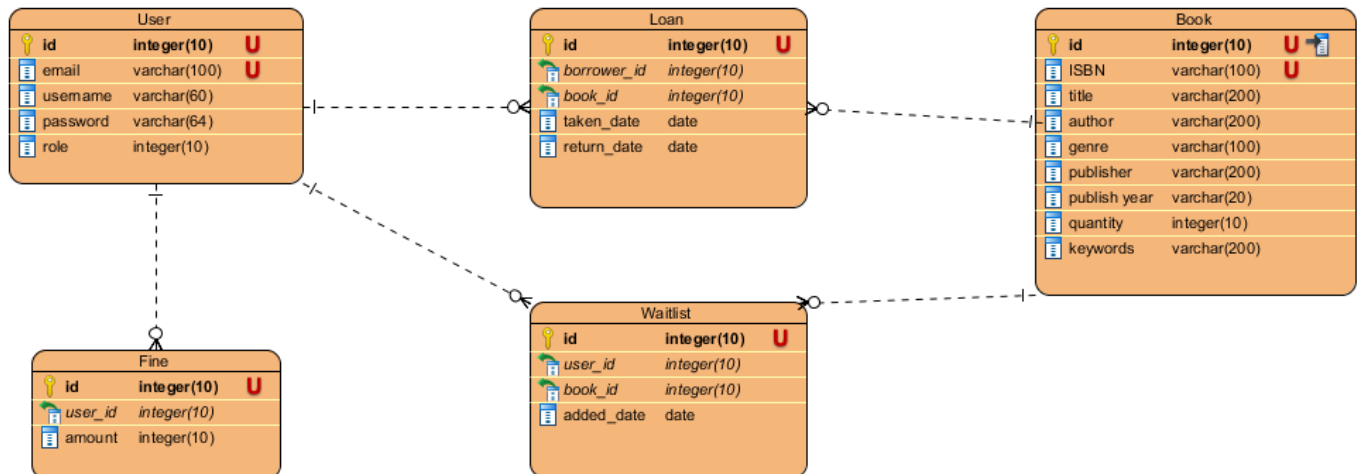Please refer to "SDD - Sequence Diagrams" to see sequence diagrams of use cases.

State Diagram of "Book" Class:

4.3   Data Model (E-R Diagram)

4.4   User Interface Design

4.4.1   Changes in UI
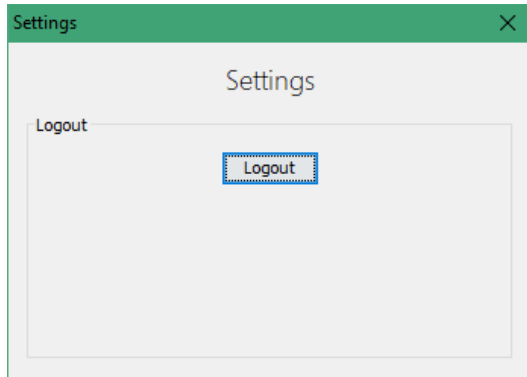
GearDialog is added to give Logout option.

Librarian: User Operations – Create Tab

Librarian: User Operations – Update Tab

In both tabs as we can see, we added "User Role" in order to specify user role. It is a combobox to prevent adding another user type.

4.4.2   Explanation of Graphical User Interfaces

For an overview of user interfaces please reference "SRS - Appendix B - GUI Interfaces" document. In this section, user interfaces will be mentioned with their respective section title.

In the graphical user interface design, we have used classical "Windows, Icons, Menus, Pointer" approach. Every window has enough controls to describe which operations it can do and what controls to be interacted in order to do operations. It uses system fonts that are easy to read. For important labels, such as window header, it uses Segoe UI Light font to make it look better.

Login Window is the first window the user will encounter. In the Login Window, the user can open "Search Books Window". When Search Books Window is opened Login Window is closed. When a user exits Search Books Window, then Login Window is opened back.

In Search Books window, user can enter 0 or more fields to search. If none is entered then it brings every

book in the database.

If a user logs into the system using Login Window, then s/he might encounter two windows. If a user is a librarian then "Main Librarian Window" will open. In the Main Librarian Window, a librarian will find buttons to open windows for Create Customer, Update Customer, Delete Customer, View Customer, Create Book, Update Book, Delete Book, View Book operations. These buttons are User Operations, Book Operations, View All Books, View All Users buttons.

If the librarian clicks to User Operations button, then "Librarian: User Operations" window will open. In this window there are 4 tabs, Create, Update, Delete, View. In Create tab, the librarian can create users. In this user interface, we added a combobox to choose user type. The librarian has to fill every field in this tab to create a user, if not then an error message will pop up. In Update tab, the librarian can update users. Librarian first needs to fill fields then press "Get User" button in order to check whether user with given information exists or not then librarian can update fields and press "Update User". In Delete tab, the librarian enters an email address of user then clicks to "Get User" button. If there is no user, then it doesn't bring user but if user exists then it shows user in the table. After that librarian clicks to user then "Delete User" button. In View tab, librarian enters information about user, then clicks to "Get User" button. If a user exists then it shows user in the table and additional book information.
In every tab, after each operation, it shows a message if it is successful or not.

If the librarian clicks to Book Operations button, then "Librarian: Book Operations" window will open. Just like User Operations window, it has got 4 tabs: Create, Update, Delete, View. It works just like User Operations.

View All Books and View All Users windows contain a table to show all of books/users.

If a user is a customer then Main Customer Window will be opened. In this window customer will find "Search Books", "My Books and Reservations", "View My Fines", "Self Checkout or Return" and "Notifications" buttons. In addition to these, there will be a summary to show day, name, fine and number of books of user.

If a user clicks to "My Books and Reservations" button then "Customer: Book Reservations Window" will open. It has got 3 tabs: Current Books, Wait List and History. Current Books currently owned books, Wait List shows books on wait list and History shows previously owned books.

If a user clicks to "View My Fines" button then "Customer: View My Fines Window" will be opened. This window will show fines in a tab and enable user to pay fines with "Pay Fine" button.

If a user clicks to "Self Checkout or Return" button then "Customer: Self Checkout or Return" window will be opened. This window has barcode field to be filled so that user can fill and press Checkout or Return button. It checks given information and pops up an error message if any.

If a user clicks on "Notifications" button then "Customer: Notification Window" will be opened. This window shows which books are available right now to make a reservation.

4.4.3   Error Prevention and Operation Messages
To get an error - success messages please refer to "SDD - Error and Success Dialogs" document. This document shows how the program provides error prevention.

4.4.4   Design Considerations
User interfaces are kept as simple as possible. Interfaces are distraction-free and minimalistic. This approach increases UX.

5.   Requirements Traceability

Please refer to "SDD - Requirements Traceability Matrix" document.

6.   Annexes

SDD - Requirements Traceability Matrix
SDD - Class Diagram
SDD - Sequence Diagrams
SDD - Error and Success Dialogs