Word Count: 1128 word

Date: 15/03/2023

1. **Introduction**

This report covers my experience in designing, developing, and testing a library management system built as a Windows desktop application using C# and Microsoft Visual Studio under the name of Insight Library, two different application was utilized during the system building stage Form (Widows form) and a service-based database. The system is designed to allow the registration as new librarian and log in to add books and client’s details throughout three MS Windows forms – Register, Login, and Work Interface, and store the data in a local database, the librarians can also add borrowing orders to the database, with unique book, client references, and the librarian username. This report covers a comprehensive analysis of the system interface design, database design, class implementation and evaluate the effectiveness, testing, and reflecting.

1. **Literature review**

Finally, after determining the idea of what the windows desktop application going to be about, too many examples were available with the same limitations, features, and functionalities in the same context of my system especially on YouTube however there was a few existing technologies that was academically reviewed such as LibraryThing, Koha, and Evergreen, they have been designed to provide amendable library management systems that has the features to manage their collection of books, clients, and borrowing orders (Brahmi and Nouali-Taboudjemat, 2019; Obeidat, 2019; Pishgooie, Zahedi and Hamzehei, 2021), in addition to that example, there is a web-based library called LibraryWorld management system which was identified as a viable option for small and medium-sized libraries and has multiple futures including the insght library fitures (LibraryWorld, 2023).

1. **Interface Design**

3.1. Registration Form:



This form allows the user who is going to be later on a librarian to register himself as a new employee through a unique username, and type a password and his full name and license ID number, however this form is not the first to appears after running the application. The login form is the first to appear as it is shown in the next drawing.

3.2. Login Form:



After running the application this is the first form that appears to the user, it will give the option to login if the user has an existing account or to create a new account as a new employee through the register hyperlink which going to open the register form that was introduced earlier.

3.3. The work interface Form:



After performing the login this session will be opened which is the last interface in the application where the librarian will be able to navigate through three buttons to manage the books in the library, clients, and borrowing orders. in each panel he will be able to fill some text boxes with the necessary information and manage them through four buttons add, update, clear, and delete. Moreover, each panel has a data grid view to display data while navigating through.

1. **Data Storage**



1. **Program Function**

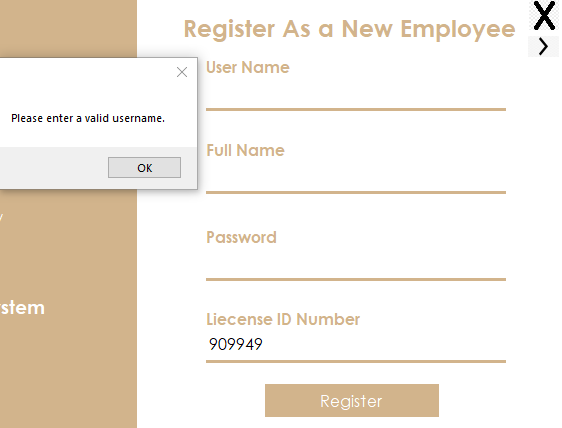
The insight library management system doesn’t have a clear structured classes as the majority of the work interface was based in one form which resulted having several event handlers in one piece of code, however the class diagram showed below explains how is could have been. The current class organization is preliminary and subject to future modifications, when there is an opportunity.



1. **Testing Plan & Standardisation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Number** | **Purpose of Test** | **Test Data** | **Expected Result** | **Actual Result** | **Comment** |
| 1 | Missing some Specific data validator after leaving a text box empty. | Displaying a Massage box for a specific empty box | All possibilities should be included, checking each empty boxes not checking box by box. | For instance, when missing two boxes, only the first message box will be displayed rather than two boxes. | (Please screenshot 1 below) |
| 2 | Testing the code flexibility while editing. | Click on view code for work interface form, zoom out the code to see a general view of the code structure. | No repeated code lines. | Connecting string variable being declared in each event handler | (Please see evidence screenshot 2 below) |
| 3 | The appropriate use of the Object-oriented programming. | Classes. | Using the OOP to have the flexibility and the scalability in the code. | Classes wasn’t used which resulted repetition in the fields and the methods. |  |
| 4 | Data layer connection with the data layer. | Database tables. | Correctly named tables simple nouns, no verbs, no plural nouns. | One table was named with verbs (Borrow). |  |
| 5 | Application Forms | Work interface form | Each panel should be in a form to have better performance. | Three main panels in one form |  |
| 6 | Meaningful attributes in the tables. | Register table. | Collect necessary data from new users. | No clear meaning for the attribute License ID Number |  |
| 7 | Running the application. | Database Connection String. | Running the application in any computer. | Need to adjust the connection string to run in other computers. |  |
| 8 | User interaction with the application. | Labels. | Steps clarification labels. | Less labels. |  |
| 9 | Resolution flexibility. | Resolution. | Performing properly in different platforms PC, Phone | Computer version only. |  |
| 10 | Further Application Functionalities and extending the idea. | Application Functionalities. | More extended ideas such as the ability to sell product, providing library memberships. | providing books borrowings only. |  |

Screenshot 1





Screenshot 2

1. **Evaluation**

The Insight library management system was developed to facilitate the efficient management of books, clients, and borrowing orders by librarians. The system utilized combo boxes to integrate relational data, which ideally should have been accomplished through classes. Upon the addition of a new book or client, the references for each automatically appear in the borrowing panel, as the book and client references are unique. It should be noted that the system was programmed with data validation coding (not a primary key, was programmed through data validation coding look at screen shoot). Additionally, the system allows for easy searching of books or clients through the utilization of search bars that provide clear explanations for the user. In summary, the development of relational data without adhering to Object-Oriented Programming (OOP) language structure may pose future editing difficulties, particularly in this beta version.

1. **Conclusion**

In summary, the Insight library management system is a Windows desktop application that facilitates efficient book, client, and borrowing order management for librarians. It uses three forms to store data in a local database and has a well-designed interface with user-friendly search bars. However, the lack of adherence to OOP language structure in the current beta version may lead to editing difficulties, and the class organization requires more structure in future versions. While the system has potential to be a useful tool for librarians, ongoing development and improvement will be necessary to keep up with changing needs.

1. **Appendices**

9.1. After adding new borrowing order, the book availability zill change to taken in the books panel and besides that after clicking on the return and delete button on the borrowing panel the book availability zill be updated automatically on the books management panel and changed to available.

9.2. the user’s name combo box used in the borrowing panel to retrieve data from; the users table is not updating automatically; the application has to be retuned so that the new user’s name zill appear in the combo box.

1. **Referencing**
2. Brahmi, F. Z., and Nouali-Taboudjemat, N. (2019) 'Design and Implementation of a Digital Library Management System', International Journal of Computer Science and Information Security, 17(8), pp. 1-6.
3. Obeidat, R. (2019) 'Library Management Systems (LMS) in Jordanian Academic Libraries: A Comparative Study', Journal of Academic Librarianship, 45(5), p. 102057.
4. Pishgooie, A. H., Zahedi, M. R., and Hamzehei, R. (2021) 'Evaluation of Open-Source Integrated Library Systems in the Middle East: Koha and Evergreen', Electronic Library, 39(3), pp. 538-553.
5. LibraryWorld. (n.d.). LibraryWorld: Library Automation Simplified. Retrieved from <https://www.libraryworld.com/>