

Automation of Library Management System to Find Books Easily

SOWJANYA REDDY MALLREDDY,

Associate Professor,

Department of Computer Science and

Engineering,

Siddhartha Institute of Technology and Sciences,

Narapally, Hyderabad, Telangana – 500 088.

RAMESH GUGULOTHU,

Assistant Professor,

Department of Electronics and

Communications Engineering,

Siddhartha Institute of Technology and Sciences,

Narapally, Hyderabad, Telangana – 500 088.

Abstract

The Library Management System was created to keep track of the receipt and issue of books in the library, as well as the details of the students. The Books Entry form is used to track new books received at the library, whereas the Student Entry form is used to track new students. When a student wants a book, it is given to him or her depending on availability of the book. Under the third option Book Issue, the book's issuance and due date are also entered into the Book Issue form. If a fee is assessed, the student is responsible for paying it depending on the number of days the book was not handed to the library. This project's major purpose is to automate the library system. The task at hand is to automate the entire application. It is saved in the database together with the stack information so that book issues can be easily controlled. It may also search for books based on availability, return books, and add new titles. To make finding easier, the books may be divided into categories. We may also obtain information on the number of books accessible in the library. The user data can be saved in the database.

1. Introduction

Manually tracing and lending books that have been returned due to making or accountability concerns is difficult. Because the borrowed books are recorded in a book, there is a good chance that this record will be misplaced or lost. The library may be unable to retrieve the borrowed books as a result of this.

The goals of this project are to implement an electronic library management system that will allow for quick and effective library service administration, as well as easy Utilities literary search and expanded access to reading materials by letting users to obtain e-books from anywhere on campus.

Information technology (IT) applications of all kinds have transformed and simplified life. In response to the rapid changes in the use of information technology, several tools, techniques, and systems have been invented and constructed. Information technology is a combination of computer science and telecommunications that allows for the creation of new systems and products to help people at work, school, and at home.

In today's world, several activities may take place at the same time and in the same place, prompting the need to integrate all processes, establish paperless environments, and provide effective management responsibilities.

2. Literature survey

The most of the objectives were met since the system allowed library staff to keep track of their clients and the resources they manage. Because of the nature of electronic storage, all data is now easy to change, making report creation straightforward. The procedure of finding reading material has been simplified since numerous criteria may be used to accomplish the task. There was no requirement for retraining other than orientation because the interfaces were user-friendly.

Presented the ISIS-software origins of the newly released ABCD-software, which has attracted the interest of many libraries and information groups in the developing countries. Many Western librarians, especially the younger ones, have never heard of ISIS, its importance, or its technical principles, therefore they strive to educate them so that they may better appraise the ABCD software's meaning for international librarianship and, why not, for their own concerns.

This research will look at the (CDS/) ISIS software as a "predecessor" to the "Free and Open Source" (FOSS) software development movement, which is gaining a lot of traction in the library and documentation industries. Even while ISIS's full dedication to this movement is new, we will demonstrate how the programme has had aspirations of being 'free' and even 'open' since its birth – which is exceptionally long in computer science traditions – by highlighting various technical elements and characteristics in this context.

3. Methodology

A programme in the Java programming language is both compiled and interpreted, which is rare. With the compiler, you first convert a programme into Java byte codes, which are platform-independent codes that are interpreted by the Java platform's interpreter. Each Java byte code instruction is parsed and executed by the interpreter on the computer. Java byte codes may be thought of as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter is an implementation of the Java Virtual Machine, whether it's a programming tool or a Web browser that can run applets. Java byte codes make it feasible to "write once, execute anywhere." Any platform with a Java compiler may compile your application into byte codes.

- **ODBC**

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for database systems providers and application developers. Programmers had to utilize proprietary languages for each database they wished to connect to before ODBC became the de facto standard for Windows programmes to communicate with database systems. From a coding standpoint, ODBC has rendered the database system essentially unimportant, which is exactly what it should be.

The benefit of ODBC from a programming standpoint is that it allows an application to utilise the same set of function calls to connect with any data source, independent of database provider. Whether the programme communicates with Oracle or SQL Server, the source code remains the same. These are only a couple of examples. ODBC drivers are available for a number of common database systems. Data sources may be created from Excel spreadsheets and plain text files.

- **JDBC**

Sun Microsystems created Java Database Connectivity, or JDBC, in order to create an independent database standard API for Java. JDBC is a general SQL database access technology that provides a uniform interface to a wide range of relational database management systems (RDBMSs). The usage of "plug-in" database connectivity modules, or drivers, allows for a uniform interface.

If a database vendor wants JDBC support, the driver must be provided for each platform that the database and Java operate on. Sun built JDBC's structure on ODBC in order to acquire greater adoption. As you learned previously in this chapter, ODBC is widely supported across a broad range of systems. Because JDBC is based on ODBC, suppliers will be able to provide JDBC drivers considerably faster than if they were designing a whole new connection solution.

The objectives specified for JDBC are critical. They'll explain why specific classes and functionalities act in the way that they do. The following are JDBC's eight design goals:

- **SQL Level API**

The designers believed that the most important task was to provide a Java SQL interface. Although it is not the lowest database interface level imaginable, it is low enough to allow for the creation of higher-level tools and APIs. It is, on the other hand, at a high enough level for application programmers to securely use it. In order to achieve this aim, future tool manufacturers will be able to "create" JDBC code and conceal many of JDBC's complexity from the end user.

- **SQL Conformance**

As you migrate from one database provider to the next, the SQL syntax changes. JDBC will allow any query statement to be given via it to the underlying database driver in order to accommodate a wide range of vendors. This enables the connection module to handle non-standard capabilities in a user-friendly manner.

4. Result and discussion

The implementation step entails meticulous planning, research of the current system and its implementation restrictions, creation of changeover techniques, and assessment of changeover methods. The proposal's implementation stage is when the theoretical design is translated into a workable system. As a result, it may be seen as the most crucial stage in ensuring the success of a new system and providing the user confidence that the system will operate and be effective.

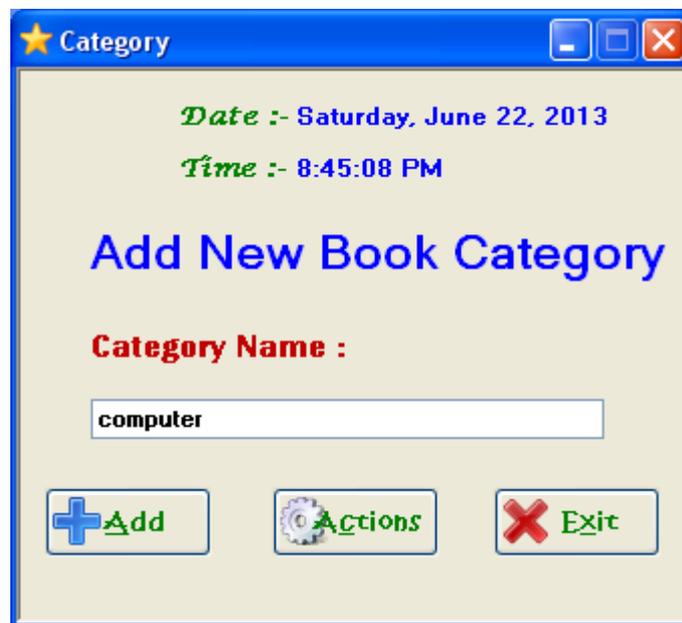


Fig 1. Adding book category

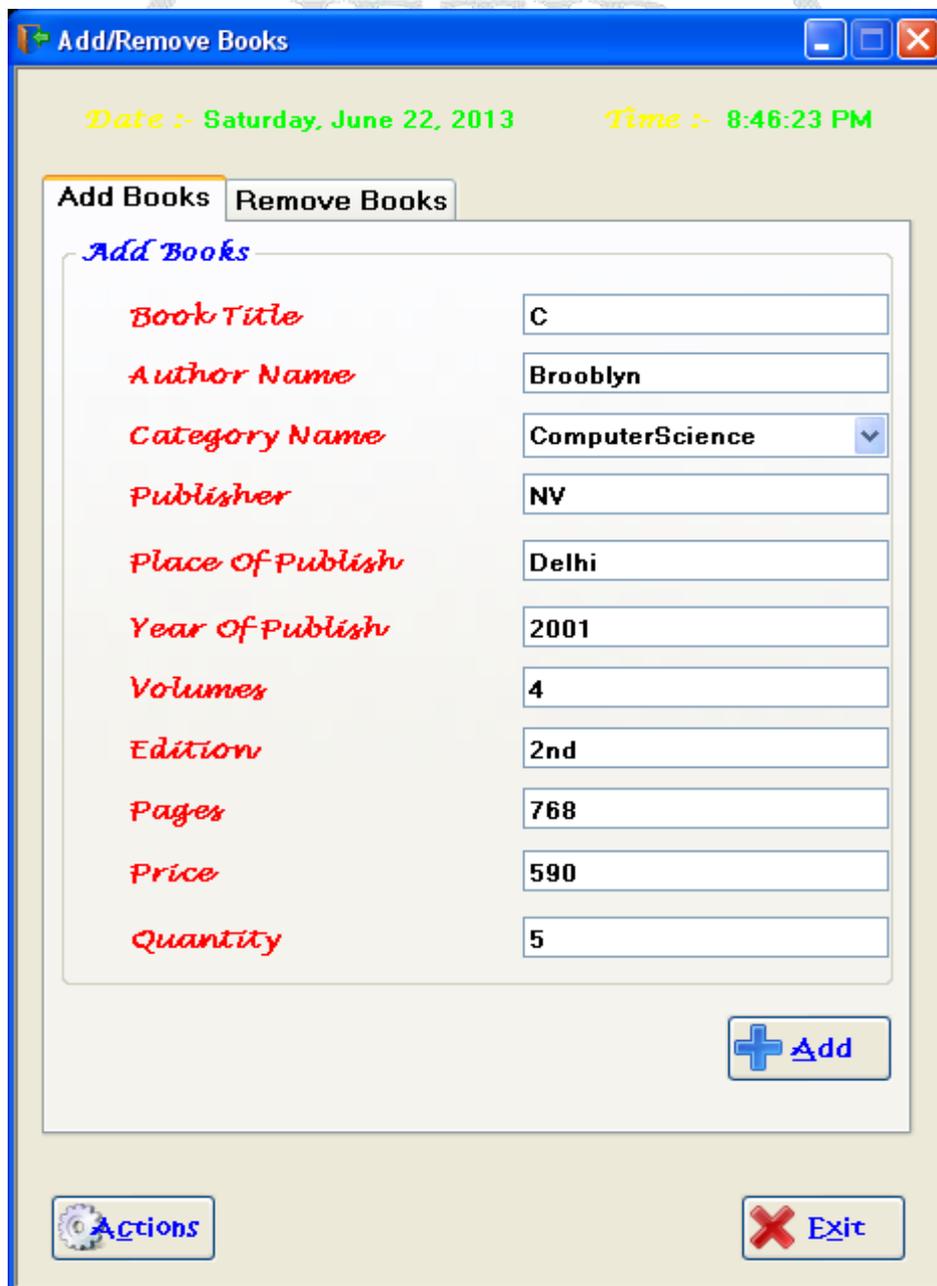


Fig.2 Add or remove books

5. Conclusion

Our proposal is only a modest attempt to meet the demands of a library. Several user-friendly coding systems have also been implemented. This package will show to be a great tool for meeting all of the organization's needs. The goal of software planning is to offer a framework for the development of software manager to produce acceptable forecasts in a short period of time at the start of the software project and should be updated on a regular basis as the project progresses advances.

References

1. Adegboye, A. M. (2010). Automation in two Nigerian university libraries.
2. Boss, R.W. (2008), “‘Open source’ integrated library system software”, available at: www.ala.org/ala/pla/plapubs/technotes/OpenSource.doc (accessed 21 February 2009).
3. De Smet, E. (2008). The ISIS-software family: from ‘Free and Open ‘to ‘Free and Open Source Software’. *Innovation*, 36(1).
4. De Smet, E. (2010). Some ISIS-Software history and technical background on the new FOSS integrated library system ABCD. *Liber Quarterly*, 19(3/4), 324-335.
5. De Smet, E. (2010). Some ISIS-Software history and technical background on the new FOSS integrated library system ABCD. *Liber Quarterly*, 19(3/4), 324-335.
6. Husain, S. and Ansari, M.A. (2007), “Library automation software packages in India: a study of cataloguing module of Alice for Windows, LibSys and Virtua”, *Annals of Library and Information Science*, Vol. 54 No. 3, pp. 146-51.
7. Jibia, M. S., Mubaraka, C. M., & Michael, O. (2013). Integrating ICT in Library Management: Design and Development of an Automated Library Management System for Cavendish University Uganda. *Innovative Systems Design and Engineering*, 4(5), 1-11.
8. Mukhopadhyay, P. (2002), “Progress of library management software: an Indian scenario”, *Vidyasagar University Journal of Library and Information Science*, pp. 1-15.
9. Ravichandra Rao, I.K. and Abideen Sainul, P. (1999), “Features of library automation software: a comparative study: library science with a slant”, *Documentation and Information Studies*, Vol. 36 No. 4, pp. 211-28.