

LIBRARY MANAGEMNT INFORMATION SYSTEM

Lect. **Magnolia Tilca**, PhD
"Vasile Goldiș" Western University of Arad
Economy Science Faculty
m_rebeles@yahoo.com

Abstract

The focus of any educational institution is the content and services of the university library. The mission of the library is to obtain, organize, preserve and update the information with the greatest possible accuracy, minimum effort and time. This requires automation of the library's operations. This paper presents a software application for managing the activity of the territorial "Vasile Goldiș" West University library. The application is developed using Visual Basic for Application programming language and using the database management system Microsoft Access 2010. The goal of this application is to optimize the inner workings of local library and to meet the requests of the institution and of the readers.

Key words: the automation of the library's management system, IT Management, VBA, MS Access

JEL: L86, M15

INTRODUCTION

The computerization of the library is currently regarded as a necessity, not as an option. The objectives of an automation project are to eliminate the routine and time consuming activities and to ensure quickly access to information. Within this context, this article describes the first step that took place towards computerization: the implementation of an integrated library automation system. This implies the existence of a database as a single file of the library. The file is managed by the appropriate software and it is accessed through local network by all services of the library. [1]

The Romanian software market develops a wide range of software products for libraries, not only local connectivity systems, but systems with global connectivity. However, developing a personalized library software system has set up the optimal solution relative to demands, time and facilities.

The technology used in the implementation of the software system is the relational database management system MS Access and Visual Basic for Applications programming language (VBA). MS Access stores data in the characteristic format of Jet (Joint Engine Technology) database Access engine. Jet, combined with the facilities of the MS Access graphical user interface and the software development tools VBA form the Microsoft Office Access.

The selection of these technologies is based on three main criteria: the MS Access application availability through software product suite Microsoft Office Professional witch no need for additional software database since the computers

have purchased this suite of products, the easy installation of the system on the work units - a single executable file with the extension .accdb which contains both the database and the forms and reports of the system, the possibility to transpose the single-user software system into multi-user system by using the data sharing property.

This software system represents the solution that allows librarians a fast and comfortable operation of activity: it provides books and magazines management, subscribers management, borrowed, returned or lost books management, easy to search by fields, keywords, project topics, it facilitates the search of a subscriber knowing the book loaned, it generates reports and statistics on the subscribers' loyalty, the load of books rate, it generates reports on the fund allocated to purchasing copies, it notes the available inventory numbers etc. The software generates statistics and reports which support the management decision process: the organization, the coordination, the training and the monitoring-evaluation of the library's services. Also, the information service module for subscribers is under development.

THE VISUAL BASIC FOR APPLICATIONS PROGRAMMING LANGUAGE

Visual Basic for Applications (VBA) is a complex programming language for application development which is integrated in the working environment programs Microsoft Visual Basic Microsoft Office. MS Access loads and installs the MS Visual Basic programming environment which is used for editing the VBA code language and also for compiling and debugging the errors of the program. The MS Visual Basic development environment, like other development environments, has the auto complete property: it suggests the programmer the methods, variables, classes, functions, etc. that are predefined by the programmer or by the development environment.

The VBA language in Microsoft Office 2010 provides a suitable framework for developing Windows applications in general, and for creating of complex and professional Access software systems in particular. VBA expands Access' possibilities due to specific features of working with object oriented programming. VBA interacts with the objects of the host application Access (tables, forms, reports, controls and modules) via the programming libraries and the API documentation (Application Programming Interface) that can be examined within the VBA development environment using the Object Browser.

The source code of the library software system is concentrated in 27 forms created in MS Access. The forms allow users to view, manage and monitor database data in a user-friendly environment. The library software system uses not only data entry forms (e.g. „Inserare Exemplare” or “Inserare Imprumut”) which provide users a fast interface to add and update data from the database, but also dialog box forms (e.g. “Cautare titlu autor”) which act as a communication interface between user and the system data. [2].

THE DATABASE MANAGEMENT SYSTEM MICROSOFT ACCESS

MS Access is a software application that stores and processes data as relational databases. It is both a data storage system and a data management system. It is easily used by thousands of users who want to effectively manage the activities, due to its understandable graphical interface [3]. Access represents the interface between developers / users and the database. The database processed by MS Access and Visual Basic is managed by Microsoft Jet Database Engine (Joint Engine Technology), which over time has become almost synonymous with MS Access.

The database of the management software system is stored in the Access file bibliotecaUBM.accdb and contains 18 interrelated tables: 7 tables of type I (tables without foreign keys), 4 type II tables (tables with a foreign key) and 7 tables of type III (tables with multiple foreign keys) (Figure 1), 15 of them being found as nomenclature in the library software system. The database fulfills the conditions of the third normalization form and respects the minimal integrity and behavior restriction.

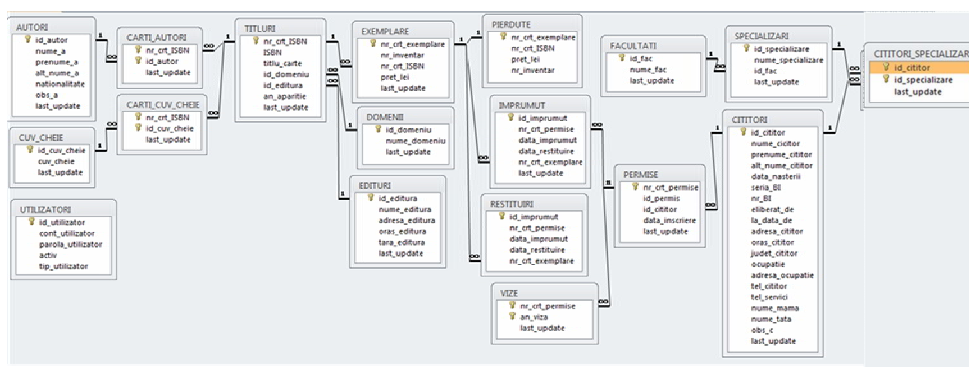


Figure 1. The entity-relationship diagram of the database

THE LIBRARY MANAGEMENT INFORMATION SYSTEM - DESCRIPTION AND FUNCTIONALITY

The software system is exclusively developed for the local university library. It is a single user database system for the administration module and a multiuser one for the loan editing module and for the subscriber information module. It runs on workstations with Windows operating systems which respect the minimum software and hardware requirements: Windows XP + SP3, CPU Intel PIV 1 GHz respectively, 1 GB RAM, 40GB HDD, LAN 10/100 for server. The workstations must be local network connected and the MS Office package must contain MS Access 2010 version.

Because the administrative structure of the organization is a simple one, the multiuser service is insured by the local network sharing option of the database folder, so all the objects of the system are shared. The bibliotecaUBM.accdb file of

the software system is stored on a shared network drive within the organization, and users can access the database file simultaneously.



Figure 2. The function of the Edit (Editare) control in the administrative module

By accessing the file bibliotecaUBM.accdb the library software system launches the login form of the type of user: administrator, librarian or subscriber. The user, according to the type, is conducted to the start form of the system, the form which contains the main commands of the menu. The administration module developed within the software system provides three main functions to the librarian: publishing, search, statistics. In turn, these functions are grouped by action the librarian takes: Nomenclatoare, Înscriere - Permis, Împrumut. The Tab Control facilitates the display of the commands of each function group on the same surface (Figure 2).

The insertion of subscribers in the database is performed by following the route of group commands from the Registration-Licence (Înscrieri-Permis). The forms involved in the process of adding new subscriber, maintain the structure and design of the form involved in the process of adding new books. The operations of borrowing and reversing a book are made by accessing the appropriate commands on the Edit (Editare) tab form the Borrow group (Imprumut). A characteristic function of the Reversions form (Restituiri) is the check of the timely reversion of the book. Optionally, the subscriber may be penalized with 1 RON for each day of delay (Figure 5). If a book is lost the software system displays the triple price of the book that should be paid by the subscriber.

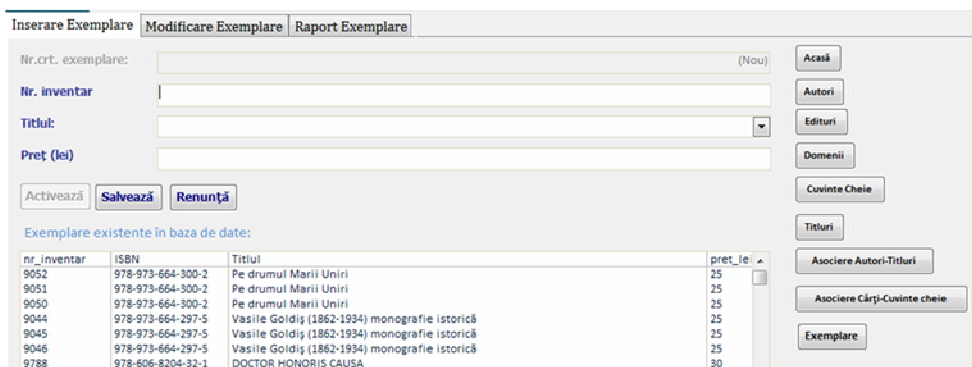


Figure 3. The edit form of the book copies

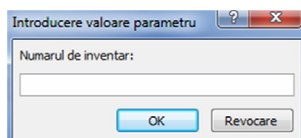


Figure 4. The dialog box with the inventory number of the book as parameter

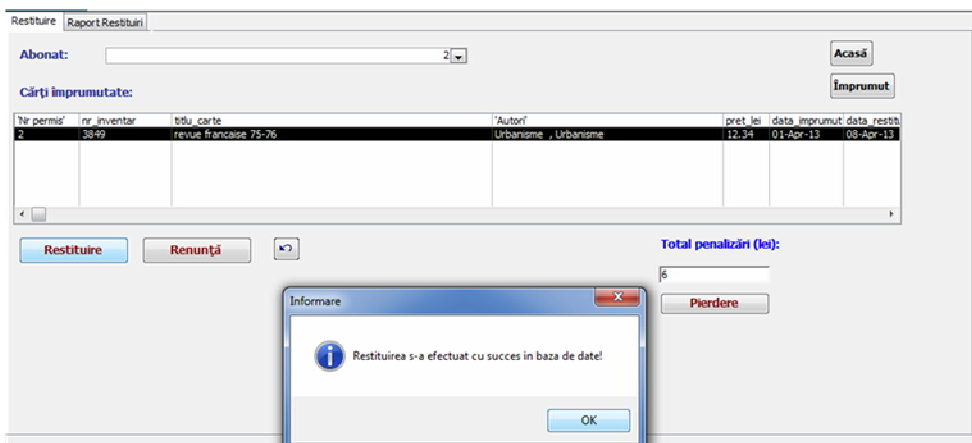


Figure 5. The penalty in the overcoming of the deadline case

The large volume of information is structured in reports and statistics provided by the software system. The reports and statistics are support in the library management because the reports offer lists of books, publishers, authors, subscribers etc. which are sorted by various criteria: academic years or calendar years, fields and so on, and because the statistics reveal, for example, the number of loans made for each book, the unborrowed books etc. This information is necessary in the decision process of achieving the success of the library activity. (Figure 6).

Statistică - Numărul de împrumuturi* efectuate de abonații bibliotecii, grupate pe domenii	
'Nr. de împrumuturi'	Domeniul
6	istoria universității
2	istorie
2	Geografia Turismului

* împrumuturile se referă la cărțile restituite

Figure 6. The statistic suggests the areas of interest to readers

A specific property of the software system is to generate the list of the unassigned inventory numbers on books. The difficulty consists in the transformation from character data type into numerical data type of the inventory number value filed (nr_inventar) (e.g. 9876/1). For this purpose we use the csng() function to convert to a Single data type. Next we determine the largest inventory number given to a book and we generate the list of numbers from 1 to the maximum value. After that we generate the list of the numbers that are not among the inventory numbers from the database.

```
Private Sub listaNrInv_Click()
' variabile:
Dim sqlMaxNrInv As String
Dim db As Database
Dim rs As Recordset
Dim maxNrInv As Single
Dim sqlTmpTabDelete As String
Dim sqlTmpTable As String
Dim i As Single
Dim sqlIns As String
Dim sqlLipsaNrInv As String
sqlMaxNrInv = "SELECT max(nr_inv_nou) as max_nr_inv" & _
" FROM (SELECT csng(val(iif (instr(nr_inventar, '/') > 0," & _
"left(nr_inventar,Instr(nr_inventar, '/')-1),nr_inventar)))"& _
"AS nr_inv_nou FROM EXEMPLARE WHERE nr_inventar is not null)
AS a"
Set db = CurrentDb
Set rs = db.OpenRecordset(sqlMaxNrInv, dbOpenDynaset)
maxNrInv = rs![max_nr_inv]
If maxNrInv > 0 Then
' Daca numarul de inventar exista atunci va trebui creata o tabela
temporara care sa fie 'populata cu valorile de la 1 la
valoarea maxima pentru numerele de inregistrare, urmand ca
' dupa ce se umple tabela, sa se faca un select pentru a extrage
numerele de inventar lipsa
sqlTmpTabDelete = "drop table tmpMaxNrInv "
db.Execute (sqlTmpTabDelete)

sqlTmpTable = "CREATE TABLE tmpMaxNrInv ( nr_inv numeric not null )"
db.Execute (sqlTmpTable)

For i = 1 To maxNrInv
sqlIns = "INSERT INTO tmpMaxNrInv (nr_inv) values (" & i &
")"
db.Execute (sqlIns)
Next
```

```

' aici avem tabelul creat cu numerele de inventar de la 1 la maxim
dupa care trebuie sa 'obtinem numerele lipsa din not in from un
select din numerele de inventar existente
sqlLipsaNrInv = "SELECT * FROM tmpMaxNrInv WHERE NR_INV NOT IN " & _
"(SELECT nr_inv_nou FROM (SELECT csng(val(iif (instr(nr_inventar, '/')" & _
">0,left(nr_inventar,Instr(nr_inventar, '/')-1, nr_inventar)))" & _
" AS nr_inv_nou FROM EXEMPLARE WHERE nr_inventar is not null) AS a )"
End If
' deschiderea raportului:
On Error GoTo Err_listaNrInv_Click
Dim stDocName As String
stDocName = "Raport lista nr inv lipsa"
DoCmd.OpenReport stDocName, acPreview
Exit_listaNrInv_Click:
Exit Sub
Err_listaNrInv_Click:
MsgBox Err.Description
Resume Exit_listaNrInv_Click
End Sub

```

Figure 7. The VBA source code that displays the unassigned inventory numbers

Another way of finding the desired information is the search process. You can get information about the existence of a book in the library that can be identified by searching a part of the title or full title, by author, by domain, keywords, and so on, you can watch the previous and current borrow situation of a subscriber (Figure 8), or you can identify who has borrowed the book sought.

The module of management of licenses of readers and of books borrowed is integrated in the administrative module (see Figure 2), so it has a similar structure. For achieving all of the objectives proposed in the library automation process, it is envisaged the development and improvement (optimization) of both modules of the information system including the web service providing access to the information offered by the library.

The screenshot shows a web-based search interface. At the top, there are three tabs: "Căutare împrumut actual", "Căutare Istoric împrumut", and "Căutare carte împrumutată". Below the tabs is a search form with the label "Introduceți numele abonatului:" and a "Căutare" button. Underneath, there is a section titled "Cărți împrumutate și restituite:" with an "Acasă" button. A table displays the search results for a subscriber named "Panait Gheorghe Alin". The table has columns for "Nr_permis", "Abonatul", "data_împrumut", "data_restituire", "Titlul", "Autori", "ISBN", and "nr_inventar". Below the table, there is a summary box stating "Numărul de cărți împrumutate: Abonatul Panait Gheorghe Alin a împrumutat 3 cărți."

Nr_permis	Abonatul	data_împrumut	data_restituire	Titlul	Autori	ISBN	nr_inventar
1	Panait Gheorghe Alin	11-Apr-13	18-Apr-13	DOCTOR HONORIS CAUSA	Man Vasile, Man Vasile	978-606-8204-32-1	9789
1	Panait Gheorghe Alin	11-Apr-13	18-Apr-13	Pro Universitaria II	Ardelean Aurel, Ardelean Aurel	973-8161-49-5	5526
1	Panait Gheorghe Alin	14-Apr-13	21-Apr-13	National Geographic mai	National Geographic, National Geogr		

Figure 8. The search form of the number of borrows made by a subscriber

CONCLUSIONS

This article is a brief overview of the integrated software system and implemented within the university library, starting at the motivation of using the selected software components in the development system and ending with the description of its most important facilities.

REFERENCES

- [1] Banciu, D., 1997. *Sisteme automatizate de informare și documentare*, București: Editura tehnică.
- [2] Beitler, B., *The Benefits of Microsoft Access Forms to Create a User Friendlier Database*, <<http://bucarotechelp.com/program/vb/89071904.asp>>.
- [3] Mocian, I., 2007. *Baze de date. Terminologie, proiectare, SQL, Access*, București: Editura Matrixrom.
- [4] Boriga, R., Huțanu, V., 2005. *Access - curs pentru liceu*, București: L&S Info-Mat.
- [5] Ribbert, U., 1998, *Managementul de bibliotecă*, București: Editura Kriterion.
- [6] databasedevco.uk, *Database solutions & downloads for Microsoft Access*, [Interactiv]. Available at: <<http://www.databasedev.co.uk/jet-database-engine.html>>.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.