



Acc. No. (FMS) T-619

GENERAL LIBRARY SYSTEM



Developed By

Amna Mahmood

Supervised By

Mr. Zaheer Aziz

**DEPARTMENT OF COMPUTER SCIENCE
INTERNATIONAL ISLAMIC UNIVERSITY
ISLAMABAD**

T-419

NA/NSC

005.74

AMG

8/1/10

DATA ENTERED 2

19-9-12

- 1 Database - Computer science
- 2 Information and retrieval system
- 3 File processing databases

* T-419

DEPARTMENT OF COMPUTER SCIENCES

INTERNATIONAL ISLAMIC UNIVERSITY ISLAMABAD

FINAL APPROVAL

April 18, 2003

This is to certify that we have read the thesis submitted by Miss Amna Mahmood. It is our judgment that this thesis is of sufficient standard to warrant its acceptance by International Islamic University Islamabad for the MSc. Degree in Computer Science.

COMMITTEE

1. External Examiner

ee
Ms. Nasim Bhatti
Director Computer Training Center
CTC, UGC, H-8, Islamabad



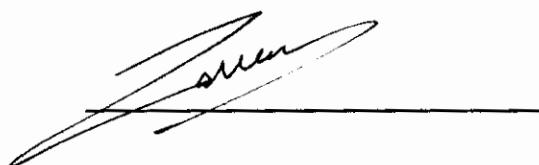
2. Internal Examiner

Mr. Sohail Iqbal
Faculty Member
Department of Computer Sciences,
International Islamic University Islamabad



3. Supervisor

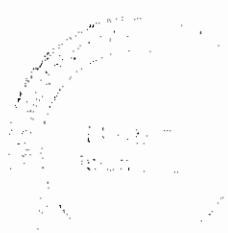
Mr. Zaheer Aziz
Faculty Member
Department of Computer Sciences,
International Islamic University Islamabad





T-419

**DEDICATED
TO
MY PARENTS**



**Dissertation Submitted To
International Islamic University Islamabad**

**As A Partial Fulfillment Of The
Requirements For The Award Of The
Degree Of**

Masters In Computer Science

DECLARATION

I, Amna Mahmood, solemnly declare that no portion of the work in this dissertation has been submitted in support of any application of any other degree or qualification of this or any other university or institute of learning. It is further declared that I have developed the software and the accompanied report entirely on the basis of my personal efforts made under the sincere guidance of my teachers. If any part of this system is proved to be copied or found to be reported, I shall stand by the consequences.

Amna Mahmood
Registration No: 312-CS-MCS/99

ACKNOWLEDGEMENTS

Thanks to '**Almighty Allah**', whose blessings empowered me to complete this project successfully. Today I owe my success to my parents, who encouraged me to gain this success by supporting me in every aspect. Also I have a good fortune of finding good teachers all over my academic period in the University.

I express my profound gratitude to my respectable teachers who guided me for every step. Especially '**Dr. Khalid Rashid**' who provided me the exact material for my project work. I would like to thank '**Mr. Muhammad Riaz**', Librarian Islamic University, Management Sciences Department, who guided me for every step of library activities. He provided me relevant material and books to understand the system.

Special thanks to '**Mr. Shahzad**' and '**Mr. Sabir**' for their cooperation in the implementation of the software.

I am obliged to '**Mr. Zaheer Aziz**', my supervisor, for guiding me in completing the dissertation and for giving me his precious time whenever I requested.

ABSTRACT

This project, 'General Library System' is a general system that is developed in such a way that it can be used for any academic library. Various facilities have been provided to the user to control acquisition and ordering of books and serials, data entry, book issue and return by using forms. Using entry forms, the user enters the data. The records can be modified using modification forms. For the validity of data various checks have been placed. The main part of system is Circulation Desk Processing. The record of books issued to any particular member is entered in the form. The system automatically calculates the fine for the overdue books.

PROJECT IN BRIEF

Project Title: General Library System

Objectives: To provide a generalized system to automate the working of library activities with maximum facilities provided to the user. The automation system provides the user every facility of data entry and manipulation by keeping track of books issued and returned by time to time.

Undertaken by: Amna Mahmood

Supervised by: Mr. Zaheer Aziz

Date Started: October 2001

Date Completed: April 2003

Tools Used: Oracle 8.05 as back-end database engine, Form Developer*6i*, and Report Developer*6i*.

Operating System: Windows XP

System Used: Pentium 4

TABLE OF CONTENTS

<u>Chapter No</u>	<u>Contents</u>	<u>Page #</u>
1. Introduction.....		1
1.1 Automation in Libraries.....		1
1.2 Need for Automation.....		2
1.2.1 Growth of Documents.....		2
1.2.2 User Services		2
1.2.3 Greater Efficiency.....		3
1.2.4 Cooperation and Resource Sharing.....		3
1.3 Project Overview.....		3
1.4 The Scope of Project		5
1.4.1 Ordering and Acquisition System.....		5
1.4.2 Cataloguing System.....		6
1.4.3 Circulation Control System.....		6
1.4.4 Serial Control System.....		7
1.4.4.1 Ordering and Acquisition		7
1.4.4.2 Cataloguing System.....		8
1.5 Objectives of the Project.....		8
2. Existing System.....		9
2.1 Overview of Library Activities		9
2.2 Technical Services.....		10
2.2.1 Criteria for Selection.....		10
2.2.2 Handling Acquisition.....		10
2.3 User Services.....		11
2.3.1 Handling Membership Categories.....		11
2.3.2 Registration of Members.....		11
2.3.3 Handling Circulation desk Processing.....		11
2.4 Drawbacks in the Existing System.....		12

3. Proposed System.....	13
3.1 Objectives of the Proposed System.....	13
3.2 Key Features of the Proposed System	15
3.2.1 Simple Data Input Facility.....	15
3.2.2 Modifying Task.....	15
3.2.3 Simplicity of Operation.....	16
3.2.4 Queries and Viewing Facility.....	16
3.2.5 Printed Reports.....	16
4. System Design.....	17
4.1 Design Principles.....	17
4.1.1 Problem Partitioning.....	17
4.1.2 Abstraction.....	17
4.1.3 Design Strategy.....	18
4.1.4 Modularity.....	18
4.2 Design Methodology.....	19
4.3 Input Design.....	19
4.3.1 Form Design.....	20
4.4 Output Design.....	28
4.4.1 Screen Outputs.....	28
4.4.2 Reports.....	29
4.4.2.1 Acquisition of books report.....	30
4.4.2.2 Monograph written by particular author report.....	31
4.4.2.3 Total number of books report.....	31
4.4.2.4 Lost books report.....	32
4.4.2.5 Overdue books report.....	32
4.4.2.6 Available books in library report.....	33
4.4.2.7 Damaged books report.....	34
4.4.2.8 Number of books issued to a particular member report.....	35
4.4.2.9 Books issued at some particular date report.....	35
4.4.2.10 Total members of ant particular type report.....	36
4.4.2.11 Member's policy report.....	37
4.4.2.12 Member's detail information report.....	38
4.4.2.13 Members left the library report.....	38
4.4.2.14 Acquisition of serials report.....	39
4.4.2.15 Serials report.....	39
4.4.2.16 Serial copies report.....	40

4.5	File Organization and Database Design.....	41
4.5.1	Database Model.....	41
4.5.2	Data Structure.....	41
4.5.3	Normalization.....	41
4.5.4	File Design.....	42
4.5.4.1	Acqbooks.....	42
4.5.4.2	Bookorder.....	43
4.5.4.3	Books.....	44
4.5.4.4	Supplier.....	45
4.5.4.5	Publisher.....	46
4.5.4.5	Member type.....	47
4.5.4.6	Member.....	48
4.5.4.7	Leaving_member.....	49
4.5.4.8	Circulation.....	50
4.5.4.9	Lost_book.....	51
4.5.4.10	Serial.....	52
4.5.4.11	Serial_copies.....	53
4.5.4.12	Acqcopy.....	54
4.5.4.13	Serial_publisher.....	54
4.5.4.14	Acqserials.....	55
4.5.4.15	Serialorder.....	56
5.	Implementation.....	57
5.1	Software Tools used.....	57
5.1.1	Oracle 8.05 as Database Engine.....	57
5.1.2	Form Developer <i>6i</i>	58
5.1.3	Report Developer <i>6i</i>	59
5.2	Form Designing.....	59
5.2.1	Data Entry forms.....	60
5.2.1.1	Form Level Triggers.....	60
5.2.1.2	Canvases.....	63
5.2.1.3	Alerts.....	63
5.2.1.4	Generation of Sequence number.....	63
5.2.1.5	Data blocks and triggers used.....	64
5.2.1.6	List of values and Record groups.....	65
5.2.1.7	Property classes.....	65
5.2.1.8	Attached Libraries.....	65
5.2.1.9	Menu module.....	65
5.2.1.10	PL/SQL Libraries.....	66
5.2.2	Leaving Member form.....	68
5.2.3	Acquisition forms.....	68
5.2.3.1	Data blocks and triggers.....	68
5.2.3.2	Program Units.....	69

5.2.4	Book Issue form.....	69
5.2.4.1	Data blocks and triggers.....	69
5.2.4.2	Program Units.....	74
5.2.4.3	LOVs and Record groups.....	74
5.2.5	Book Return Form.....	75
5.2.5.1	Canvases.....	75
5.2.5.2	Screen Design.....	75
5.2.5.3	Data blocks and triggers.....	75
5.2.5.4	Program units.....	76
5.2.5.5	Items Functionality.....	76
5.2.6	Modify forms.....	78
5.2.6.1	Canvases.....	78
5.2.6.2	Screen Design.....	78
5.2.6.3	Data blocks and triggers.....	78
5.2.6.4	Program units.....	80
5.2.6.5	LOVs and Record groups.....	80
5.2.6.6	Property Classes.....	81
5.2.7	Serial form.....	81
5.2.7.1	Screen Design.....	81
5.2.7.2	Data blocks and triggers.....	81
5.2.8	Search forms.....	82
5.3	Report Designing.....	82
5.3.1	Development of report using wizard.....	83
5.3.1.1	Style of report.....	83
5.3.1.2	Query type and Data.....	83
5.3.1.3	Selecting the fields.....	84
5.3.1.4	Specifying totals.....	84
5.3.1.5	Labels and Templates.....	84
5.3.2	Editing in Live Previewer.....	84
5.3.3	Creating Data mode.....	84
5.3.3.1	Queries.....	85
5.3.3.2	Columns.....	85
5.3.3.3	Groups.....	85
5.3.3.4	User Parameters.....	86
5.3.4	Program Units.....	86
5.4	Hardware and Operating System Requirement.....	88
6.	Testing.....	89
6.1	Testing Objectives.....	89
6.2	Testing Criteria.....	89
6.2.1	Operability.....	90
6.2.2	Usability.....	90
6.3	Testing Menu.....	90

6.4	Testing Data entry forms.....	91
6.5	Testing Modify forms.....	91
6.6	Testing Acquisition forms.....	92
6.7	Testing Issue book form.....	92
6.8	Testing Return book form.....	93
6.9	Testing Reports.....	93
Appendix A. User Guide.....		94
A.1	User Validation.....	94
A.2	Main Screen.....	95
A.3	Menu at Main Screen.....	96
A.4	Books Entry form.....	97
A.5	Modify Books detail form.....	98
A.6	Modify Books detail form(On clicking Select from list).....	99
A.7	Modify Books detail form.....	100
A.8	Leaving Member entry form.....	101
A.9	Leaving Member View/Modify form.....	102
A.10	Lost/Damaged books entry form.....	103
A.11	Lost/Damaged books modify form.....	104
A.12	Member type Policy entry form.....	105
A.13	Member type Policy View/Modify Form.....	106
A.14	Publisher Details entry form.....	107
A.15	Publisher Details modify form.....	108
A.16	Supplier Details entry form.....	109
A.17	Supplier Details modify form.....	110
A.18	Books Acquisition form.....	111
A.19	Order Books form.....	112
A.20	Search By Author form.....	113
A.21	Search by Subject form.....	114
A.22	Search by Title form.....	115
A.23	Issue book form.....	116
A.24	Return book form.....	117
A.25	Serials Entry form.....	118
A.26	Serials Entry form(On clicking button 'Add new serial').....	119
A.27	Serials Entry form(On entering title/selecting from list to add a copy).....	120
A.28	Serials Acquisition form.....	121
A.29	Serials Order form.....	122
Appendix B. ER Diagram and Data flow Diagram.....		123

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

In this chapter I will discuss the area of knowledge for which the project is developed. Next the need of the project to be developed will be mentioned in which description is given for why the project is selected for development. Then comes the scope of the project in which the problem is defined and major modules to be developed are discussed. In the end objectives of the project will be provided in which point from Need of Project and Scope of Project are summarized such that the goals become clear.

1.1 Automation In Libraries

Since the advent of the term automation in 1936, plethora of definitions is found in library literature. Sometime the terms mechanization and automation looked overlapped, although there is a difference of one handling problems of bulk, weight and distance at replacing muscle and movement with machines. Automation is the name given to an automatic system of working. The meaningful definition is examined hereunder.

According to *Encyclopedia of Library and Information Science*, “Library automation is the use of automatic and semiautomatic data processing machines to perform such traditional activities as acquisitions, cataloguing, and circulation. Although these activities are not necessarily performed in traditional ways, the activities themselves are those traditionally associated with libraries; library automation may thus be distinguished from related fields such as information retrieval, automatic indexing and abstracting, and automatic textual analysis.”

Linguistic purists have argued rightly that the term “*automation*” applies more correctly and narrowly to automatic process control, and indeed this way historically the first use of the term. The broader meaning, however, has had the sanction of widespread usage for a number of years, and “*library automation*” is now by far the most commonly used term for mechanization of library activities using data processing equipment. [1]

The general characteristics of an automated system are:

- i. The operations / processes are carried out automatically.
- ii. Avoids or reduces human action and thus saves labor.
- iii. It accelerates efficiency and speed in operation.
- iv. It increases accuracy and proficiency.

1.2 Need For Automation

Various factors have contributed to bring about change from conventional to automated library operation. With the advent of time it has been made necessary to move towards advance and active world. Broadly speaking the main reasons behind this change is as under:

1.2.1 Growth Of Document

The amazing growth of documents especially in areas of sciences, social sciences and technology in the form of books and non-book material forced the library planners to utilize new technology for the organization of information since the traditional library methods failed to cope with the task.

1.2.2 Users Services

Heavy influx of documents added new dimensions to user's needs. It is only the computers, which can ensure improved and quick service. For example the manual work of housekeeping such as acquisition, cataloguing, circulation, serials control etc. can be done with greater speed and efficiency with no arrear and backlog kept pending. Automated cataloguing provides variety of access points to a greater volume of information. Online catalogue has made it possible for a reader to press the key of the typewriter like terminal and have access to large bibliographical databases and obtain information in whatever field. It can be viewed on screen and if it is needed a printout can also be obtained. Modern services such as current awareness services and selective dissemination of information can also be introduced efficiently. [1]

1.2.3 Greater Efficiency

The workflow in the library may be made more rapid, more systematic and efficient with the help of the computer. The records in the computerized store are more accurate, more reliable and more accessible than the manually prepared records. All sorts of housekeeping jobs and information works can be performed with more speed, accuracy and greater efficiency. [1]

1.2.4 Cooperation And Resource-sharing

No single institution, however resourceful can now collect, organize all documents available all over the world even in a specific field. Nor any one organization can collect all the documents in the areas of interdisciplinary and multidisciplinary subjects. During the last two decades there has been an increase in cooperation and resource-sharing on the part of libraries by developing computer-based systems. It has two advantages: (a) *access to more documents and information source due to coordination and cooperation of many libraries in a system* and (b) *less expenditure, because the total cost is shared by a number of libraries in the system*.

The advantages of library automation are far-reaching. Housekeeping jobs aside, the facilities of time-sharing and online services have made library automation more effective for efficient and appropriate service. Today only computerized library can take part in computer networking and information networks at national and international levels. It is only way in which efficient standardized and faster service is assured, besides access to a world of information and data. [1]

1.3 Project Overview

The project titled “General Library System” deals with the computerized Library Information system. The system will automate the existing manual preparation of reports, forms, storage and processing of information and records. The automated library management system will be established as an essential tool in the support of effective

customer service, stock management and, in general, management of services offered by libraries.

The automated library system has two aspects namely housekeeping routine and information storage and retrieval. The house keeping routines include acquisition and ordering work, cataloguing, circulation control, serial control, and keeping of records.

The main module of the project is circulation system. It keeps track of the member registration and relevant dues. In addition to this the process also manages the circulation of the books i.e. Books issuing, return, sending notices etc. including supervising fines. All the members are divided into certain predefined categories and the memberships are allocated according to these categories.

The information service comprises generation and collection of information, information retrieval, current awareness services, computerized database, information transfer and distribution etc. The system will be developed as an integrated system in which all subsystems such as acquisitions, circulation, cataloguing are interrelated and share a common database. An integrated system is one in which multiple functions are supported by a single bibliographic file. In this way the system will be reliable and free of errors, as there will be no possibilities of overlapping of records. [1]

The diagram representing the integrated system is shown in figure 1.1.

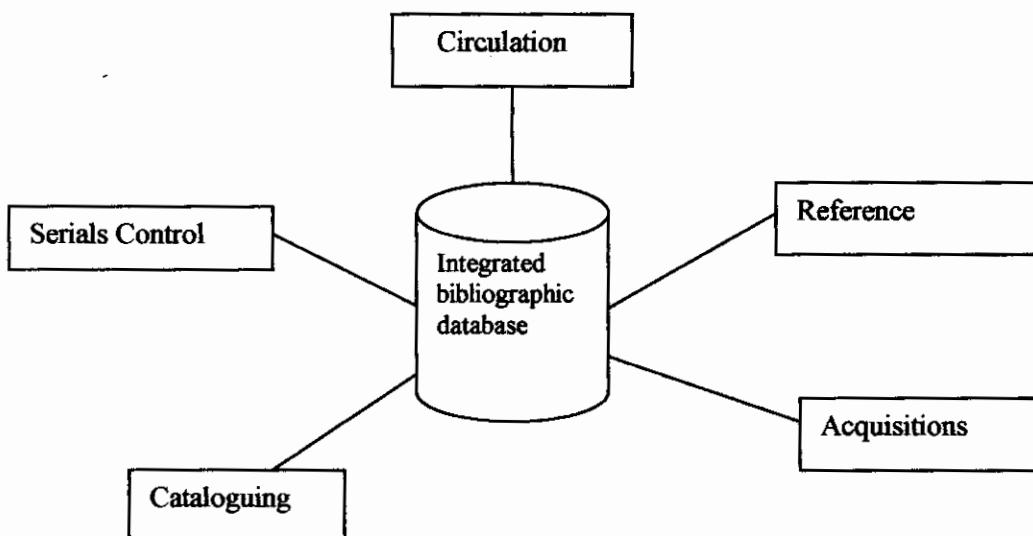


Figure 1.1 : The Integrated System

1.4 The Scope Of Project

The scope of project deals with the automation of various subsystems of the library activities. It includes the ordering and acquisition of various books, cataloguing system; circulation system and serials control system.

1.4.1 Ordering and Acquisition System:

It is a relatively simple clerical process, where similar operations are applicable to all categories of library. The system comprises the following activities:

- Receiving records of items for acquisition.
- Ascertaining whether those items are already in the library or in order processing.
- Making input of order in specified format.
- Checking whether orders are overdue and if so, chasing the supplier and sending reminders.
- Accounting under fund heads and sending bills for payment.
- Maintaining record files for items on order.
- Accessioning of items received.
- Updating the whole record to ensure latest position time to time.

Books are also received through gifts or exchange. In this case, ordering, chasing, account and billing functions will be eliminated. But these items have to be recorded on receipt and accessioned accordingly.

1.4.2 Cataloguing System:

When the item is received, the order records become the basis of catalogue records. Title, author, publisher, date of publication, corporate author, technical report numbers etc have all been keyed in at the point of ordering.

System will prepare a cataloguing worksheet for professional cataloguer containing all information entered up to the point of cataloguing. In order to complete cataloguing process, the librarian will provide subject heading, assign call number in specific format and provide other important information like author name, date of publication, publisher-place and name, ISBN/ISSN, notes, accession number etc. [1]

The system has following features:

- The system is very comprehensive in its search facility. The operator or cataloguer has to add the records and search the record in database. Search can be done either by call number or by author name.
- Librarian or operator performs this operation. For the members of library, the manual card catalogues are there to consult by themselves.
- System catalogue is searchable through variety of access points greater than those available through card form catalogue.
- Retrieves information from local library file, and if not successful locally, retrieves information from other libraries files.
- Provides instructional help.

1.4.3 Circulation Control System:

Circulation of books and other materials is an important part of library public services. It manages the circulation of books for example Books issuing, return, and sending notices. The objectives of circulation control system are summarized as follows:

- To provide the facility for registration of members.
- To issue borrower library cards.

- To get information about books to be borrowed.
- To identify borrowers by particulars such as name, address, borrower number, category, whether membership is lapsed, present status, borrower signature etc.
- To identify document to be borrowed such as call number, ISBN, other identification numbers, author, title, edition, year etc.
- To record the borrowing with particulars, such as the date of loan, date of return, to whom it is loaned, any restriction regarding the loan, category of loan etc.
- To keep records of number of books on loan to individuals.
- To check overdue books.
- To update record file.
- To prepare overdue and recall notices (prepare manually).

X

1.4.4 Serial Control System:

Serials include periodicals, newspapers, annuals, journals, memoirs, proceedings, and transactions etc of societies. It is one of the most difficult housekeeping operations to be automated mainly due to the unpredictable nature of serial publications. [2]

Serial control systems are indeed distinct from monograph systems in a number of ways. A monograph is a single entity at hand whereas serials are published at regular or irregular intervals; their contents usually vary from issue to issue. An article as a single bibliographic unit may be published in more than one volume. Secondly an issue of a serial may not be a bibliographic unit in itself but a bibliographic unit embodied in a particular article comprising more than one issue or volume.

The integrated serial control system features all the three of subsystems already encountered for monograph namely:

1.4.4.1 Ordering and Acquisition:

The system is used to control the selection, ordering and checking in of serials, payments, and chasers when indexes or issues fail to arrive. Ordering must contain records of renewal dates and subscription levels, and processing must include the capacity for scanning records to check for renewal dates. The acquisition function may be divided into receiving

and claiming. A further function also linked to the accession function is subscription control. In order to fulfill this function, the master record must include the details of, when subscriptions are to be renewed. [3]

1.4.4.2 Cataloguing System:

It keeps the record of stock. Catalogue formats, orders, processes etc for serials are fundamentally similar to those for monographs. One difference is in adding of changed and variant titles. Reference is provided between linked titles.

1.5 Objectives Of The Project

The project is the automation of a manual library system so the main purpose of the project development is to provide the better and reliable system to the end users. Some of the objectives describing the project are as under:

- The important objective of the system is to automate the existing manual system of member registration, data entry and circulation, so that the system strives for efficiency and becomes reliable.
- To provide the accurate and efficient information to the user.
- To decrease the possibilities of errors by making the system perfect.
- To overcome the problems faced by librarians in technical and user services.
- To provide the fast and efficient storage and retrieval of data to the user.
- To make the method of searching books and materials in the library easy and quick.
- To help members use new information technologies to provide equitable and effective access to information.

CHAPTER 2

EXISTING SYSTEM

2. EXISTING SYSTEM

In this chapter we'll discuss the existing working system and the terminology and concepts needed to understand the project. The system underdevelopment is the replacement of existing manual system so the library activities done manually will be discussed in such a way to understand the system.

2.1 Overview Of Library Activities

Libraries are collections of books, manuscripts, journals, and other sources of recorded information. They commonly include reference works, such as encyclopedias that provide factual information and indexes that help users find information in other sources; creative works, including poetry, novels, short stories, musical scores, and photographs; nonfiction, such as biographies, histories, and other factual reports; and periodical publications, including magazines, scholarly journals, and books published as part of a series. [4]

Prior to this phase of computerization, all the library functionality was either non-existent or handled manually by the staff. In the manual system, the task of keeping records is very difficult and time consuming. Receiving the books and serials, then selecting them, and then preparing the registers for keeping the records is really hard thing. In this way large amount of registers are preserved for a long period of time. Searching the details of any particular book is nearly impossible. The working staff had a major problem of preparing reports and keeping track of memberships. Apart from this, the existing manual system has wide possibilities of errors.

There are two major functions of libraries namely technical services and user services.

2.2 Technical Services

One of two major functions of libraries, technical services include processes for acquiring, arranging, indexing, and storing the collection.

2.2.1 Criteria for selection

The output of published materials, in all forms, is so vast that no single library, not even the largest, can hope to acquire everything; even in relatively specialized fields, some selection has become necessary, and most libraries have an explicit selection policy. The basic principles of selection vary little among different types of libraries, inasmuch as they derive directly from the known interests of the users. Practice is another matter and varies according to the types of user. A national library aims to hold at least one copy of all the publications of its own country and to have a good representation of foreign works, many of which may be obtained through exchange agreements with other national libraries. University, college, and school libraries relate their choice of acquisitions to the programs of teaching and research in their institutions; the academic level of the material naturally varies according to the level of the student population. An elementary school will hold a good selection of books written for children, but a university will tend not to. Many university libraries try to maintain a relatively complete coverage of the reports issued by government and other research establishments. [4]

2.2.2 Handling Acquisition

The books and serials acquisition is handled manually by making a separate register having the details of the vendors/suppliers and publisher, from where the books has been arrived. After the selection, the book or serial is marked rejected. For this purpose the record of the particular book is searched that is very difficult task.

2.3 User Services

The second of the two main functions of libraries is directed at actively exploiting the collection to satisfy the information needs of library users. [4]

2.3.1 Handling Membership Categories

In the existing system there is no plan for managing membership categories. The circulation department handles all the members manually. All memberships are allotted under these membership categories. All conditions of dues, fines, no of books issued, days for fine validity etc are handled by maintaining a separate register in which all the information is stored. The register is consulted again and again as new members arrive. This manual system may pose serious problems for the circulation department.

2.3.2 Registration of Members

The members are registered for the specific category already registered. Under no circumstances a member can be registered under a category that is not registered previously. The complete information of the member arrived is kept in a separate register. The user has to rely on the thinking and memory for some issues regarding membership security money. Then a membership card is issued to the member for a particular period of time. For this membership card the member is issued any material from the library.

2.3.3 Handling Circulation Desk Processing

Circulation is the process deals with the issuance of books to members. The book is issued for the membership card. The return of the books and handling of fine for the overdue books face some problems. It is difficult keeping track of overdue books for a long period of time and then calculating the fine. As well as the preparing notices for the return of overdue books is handled manually.

2.4 Drawbacks In The Existing System

The existing system is inefficient as it takes a long time for handling the records of the books, serials, as well as the members and the book issuance, return and other procedures.

- According to the library rules only the authorized persons should have access to the relevant information. But the existing system is not secure enough as the registers and the record books are present in the same library. The unauthorized persons can also change the information.
- No statistical information is kept for decision-making.
- Preparing the detail reports about the total no of books in the library is nearly impossible in the existing system.
- The processes in the library system take a long period of time.
- No backup of data available for emergency purpose.
- The manual system may face errors at any time.
- The manual system is slow, so the notices and the orders to be placed are not handled in time.

CHAPTER 3

PROPOSED SYSTEM

3. PROPOSED SYSTEM

This chapter proposes a computerized system and discusses suitable strategy to develop the system according to the requirements of end user. The new proposed system indicates the desired system accurately in data processing terms keeping in view the Graphical User Interface for data entry and retrieval that would increase the efficiency manifold. The users will be provided the interactive system with less human effort.

3.1 Objectives Of The Proposed System

The project is the automation of the existing manual system. These objectives listed below encompass the proposed system.

- **Efficient And Reliable**

Contrary to the manual library system where there are many registers for the working of the library, which includes members entry register, monograph entry register, circulation register etc., the new system strives for reliability as well as efficiency and to ensure that these two criteria are met for the system, special care has been taken for application design and consistency. The formatted information are presented in the forms of reports, thus no need to scan several registers.

- **Graphical User Interface**

The system should provide an easy interface for entry and retrieval of information. Therefore all efforts have been utilized to make the system as user friendly as possible and sufficient consideration has been given to the input screen design using forms. The proposed system keeps the data entry simple and consistent with the manual system.

- **Managing Database**

The data record of the members and books will be kept in the database so that the unique number of records should be maintained without any redundancy and the generation of reports could be made possible.

- **Fast Retrieval**

The proposed system will provide the speed and efficiency, which are the main factors in a computerized system design. Using the RDBMS tool the retrieval time is reduced.

- **Handling Errors**

In order to insert or retrieve the correct data and giving the possibility of error free manipulation, the system provides the data validation checks. These checks are provided in the application modules while proceeding towards entry, processing, deletion and updation.

- **Compatibility**

The front end Application System (Developer 6i) is compatible to access any backend database through ODBC (Open Database Connectivity). So it is reliable to use Oracle (RDBMS), SQL Server, MS-Access or any other RDBMS Server. At present the back end database is Oracle 8.05.

- **Data Security**

Data required for management is highly valuable. Therefore, reliability of the proposed system is secured by providing a unique user name and password to each user of the system.

- **Quick Approach**

The user is provided a menu system so that the approach to different forms and reports is made easy and quick.

- **Reports**

Automatic generation of reports is made possible in the proposed system so that the records are arranged easily and the printouts could be taken whenever needed.

- **Searching**

As most important objective of the library system, the system querying facility is implemented such that the system will provide ease for the library staff in lending and searching the books. The books can be searched in three different ways, *By Title*, *By Author* and *By Subject*.

3.2 Key Features Of The Proposed System

Capacity, access, speed and cost of system are the parameters of main concern for the management of the system. The system has been proposed in the light of above mentioned objectives. It is capable of carrying out the following.

3.2.1 Simple Data Input Facility

The proposed system will work in the same way as the present system is working with some changes in order to make process more simple and effective and to avoid double entry of any data as well as any irrelevant data. Accession number will be generated automatically as the book is entered in the database by using forms, whereas serial number will be generated for the serials. Thus making data input much simpler and easy.

3.2.2 Modifying Task

In the present system, editing or modifying any material is very difficult and time consuming. To overcome the difficulties, in the proposed system all the circulations will be carried out by the computer itself in order to avoid error, on the basis of correct input data. To modify any particular record searching will be provided either by using List of Values or

by entering the identification number. It will save time of user and error possibilities can be reduced.

3.2.3 Simplicity of Operation

The proposed system will have self-explaining menu-driven facility that will be easier to operate. Screen will guide the operator in performing different tasks. There will be messages for the user about the purpose of the option he is going to use. Validity checks will also be there for error detection. On returning the book the fine will be calculated by the system automatically, if the book is overdue.

3.2.4 Queries and Viewing Facility

One of the best advantages of the computerized system is its use in real time. Queries will also solve the problems of user to look for particular information at any time on screen rather than to search in files or registers of even outdated information whenever needed on the screen.

3.2.5 Printed Reports

Another important feature of the system will be the ease of getting printed reports at once whenever needed. The notices will be prepared for the members if the books issued to them are overdue. These reports will give the complete list of entries with not as much of time.

CHAPTER 4

SYSTEM DESIGN

4. SYSTEM DESIGN

The purpose of the design phase is to plan a solution of the problem specified by the requirements document. This chapter shows the design principles, methodology and strategy adopted for the system under development. This phase is the first step in moving from the problem domain to the solution domain. System design aims to identify the modules that are in the system, the specifications of these modules, and how they interact with each other to produce the desired results.

4.1 Design Principles

Software design is both a process and a model. There are basic guiding principles that are employed to produce the design of a system. These principles are concerned with providing means to efficiently handle the complexity of the design process. Following are the design principles.

4.1.1 Problem Partitioning

For solving large problems, the basic principle is the time-tested principle of “*divide and conquer*”. For my software design, therefore, the goal is to divide the problem into manageable small pieces that can be solved separately. However, the different pieces are not entirely independent of each other, as they together form the system. The different pieces cooperate and communicate in order to reduce the complexity. [5]

4.1.2 Abstraction

Abstraction is a tool that permits one to concentrate on a problem at some level of generalization, without worrying about the details of the implementation of the component. Abstraction is an indispensable part of the design process and is essential for problem

partitioning, for existing components as well as components that are being designed. When partitioning was used during design, the design activity focused on one part of the system at a time. Since the part being designed interacted with other parts of the system, a clear understanding of the interaction was essential for the properly designing the part. For this, abstraction was used. [5]

4.1.3 Design Strategy

Depending upon the nature of the system components, hierarchy of a system is designed. There are three different types of approaches possible namely top-down, bottom-up and mixed.

Keeping in view the natural requirements of General Library System, it is composed of three major departments/ activities i.e. purchase and acquisition, cataloguing/ data entry and circulation desk processing. These three major activities have been refined further; hence in this case top-down designing strategy has been adopted. The top-down approach starts from the highest level of the hierarchy and proceeds to the lowest levels, hence resulting in some form of stepwise refinement. [6]

4.1.4 Modularity

System modularity is important in order to achieve portability. It is the single attribute of software that allows a program to be intellectually manageable. And a module is a reusable component of a design and program. It is a program unit that is discrete and identifiable with respect to compiling and loading. Each module of the system obeys all of designing objectives and principles. [6]

4.2 Design Methodology

Creating software system design is the major concern of design phase. The aim of design methodology is not to reduce the process of design to a sequence of mechanical steps, but to provide guidelines to aid the designer during the design process. The methodology I have chosen is “Structured Design Methodology”.

Structured design methodology (SDM) views every software system as having some inputs which are converted into the desired outputs by the software system. It is a data flow-based methodology. The approach begins with a system specification that identifies input and outputs and describes the functional aspects of the system. The system specification, then are used as a basis for the graphic representation – data flow diagram (DFD) – of data flows and processes.

System design is a process of developing specifications for the candidate system that meet the criteria established in system analysis. A major step in design is the preparation of input and the design of output reports in a form acceptable to user. [7]

System design consists of following steps.

4.3 Input Design

Input design is the process of converting user-originated inputs to a computer-based format. Inputs are designed from multiple points of views.

- The goal of designing input data is to make data entry as easy, logical, and free from errors as possible.
- Inputs were placed carefully because of time, effort and cost involved converting raw data to usable information because the constraint on input and variation in input volume is affected by design description.
- Data needed for new management system, was determined and formatted to be useful to produce desired outputs. [7]

4.3.1 Form Design

Since an input form plays a very important role in data entry phase, so these forms should be easy to understand and use. Keeping all this in view, fourteen forms have been designed for entering different type of transactions with GUI and follows the standard formats. The forms have been designed having two types of modes.

- Data Entry Mode
- Modify Mode

4.3.1.1 Data Entry Mode

In data entry mode the forms are in normal mode having text boxes empty for data input. The data is simply entered and saved. The features that facilitate and ease the user are:

Password

The General Library System checks the password before entering into the project, the purpose is to allow only authorized persons in the system.

List of Values

User can select the values from the list of values instead of input it, like the publisher number, supplier number etc.

Help

User can have help about the function keys as well as about the buttons.

Validation Checks

Various checks of uniqueness, avoid of duplication of the primary key, date validation, have been imposed in the General Library System forms.

4.3.1.2 Modify Mode

In Modify Mode the forms are in search mode, the user can enter the data to be modified either by entering the id or by searching from the list of values. The complete detail of the record will be available to modify. The features related to this mode are as follows:

Search>Show

User can search a specific record with this option.

Delete

User can delete a record provided it that does not have its related transactions in some other form.

List of Values

The lists of values are provided to the user for searching the record to be modified.

The forms are as below

➤ Books Acquisition Form

It will be used for entering the books reaching library for selection by any means. The format of form contains the following fields.

- Title
- Author
- Predate
- Publisher id
- ISBN
- Supplier id
- Acquisition type
- Source
- Rejected

➤ Books order Form

It will be used to keep track of the books ordered and received by the library with the payment details. The fields designed for the form are as follows.

- Order number
- Title
- Author
- Supplier
- Number of copies

- Order Received
- Individual price of book
- Invoice number
- Payment made

➤ Books Form

It will keep the records of all books in the library. A unique accession number will be allotted to each book for identification. The fields related to the form are:

- Accession number
- Publisher id
- Source
- Supplier number
- Subject
- Title
- Subtitle
- Author
- DDC number
- Cutter number
- Edition
- Publish date
- Publishing place
- Number of volume
- ISBN number
- Pages
- Price
- Acquisition type
- Invoice number
- Status
- Reference
- Shelving location
- Remarks

➤ **Publisher Form**

It will be used to record and introduce the publishers. The fields associated with the forms are as follows.

- Publisher id
- Publisher name
- Address
- Phone
- Email
- Fax number
- Web URL

➤ **Supplier Form**

It will be used to record and introduce the suppliers. It will have the following fields:

- Supplier number
- Supplier name
- Address
- Phone 1
- Phone 2
- Email address
- Web URL

➤ **Lost Book Form**

This form will be used to keep the record of books lost or damaged by members or library authority. It will contain the fields:

- Lost/ Damage by
- Accession number
- Member id
- Lost date
- Status
- Replacement
- Book's original price

- Amount paid
- Concession
- Remarks

➤ Member Type Form

It will be used to record the categories of members and the information related to the membership category regarding the circulation desk processing. The fields it contain are as follows:

- Type number
- Type name
- Fine per day
- Days of circulation
- Maximum books allowed
- Maximum fine
- Security
- Days for validation of fine after due date

➤ Member Form

It will keep all the complete information about the members related to every membership type. This form is designed to have following fields:

- Member id
- Type number
- Name
- Father name
- Address
- Membership expiry date
- Email
- Phone
- Date of birth
- Security
- Concession

- Gender
- Member status
- Remarks

➤ **Leaving Member Form**

This form will record the members going to leave the membership. The fields it contain are as follows:

- Member id
- Type number
- Leaving date
- Security deposited
- Refund
- Amount
- Reason

➤ **Serials Acquisition Form**

It will be used for entering the serials reaching library for selection by any means. The fields related to the form will be:

- Title
- Subtitle
- Predate
- Author
- Publisher id
- ISSN
- Publication type
- Supplier number
- Acquisition type
- Source
- Rejected

➤ **Serials Order Form**

It will be used to keep track of the serials ordered and received by the library with the payment details.

- Order number
- Title
- Publisher id
- Publication type
- Supplier
- Number of copies
- Order received
- Receiving date
- Invoice number
- Payment made

➤ **Serials Form**

It will keep the records of all serials and their copies in the library. This form will be designed to store both the serials as well as their copies. The fields of the form in serial's block are:

- Serial id
- Publisher id
- Language
- Entry date
- Title
- Subtitle
- ISSN
- Publish date
- Publication type
- Frequency of publication
- Subscription date
- Expiry date
- Renewal date

The fields in the 'Serial copies' block are as follows:

- Serial number of copy
- Publish date from
- To date
- Source
- Supplier number
- Volume number
- Remarks
- Acquisition type
- Purchase price
- Invoice number
- Status
- Shelving location
- Notes

➤ Books Issue Form

It will be used for entering the books issued along with the due date. The form will have the following fields:

- Accession number of book
- Member id
- Member type number
- Issue date
- Due date

➤ Books Return Form

This form will enter the books returned to the library with the calculation of fine. Some fields will be same as of Issue book form just for the view the book whose return date is going to be entered.

- Accession number
- Book title
- Member id

- Member name
- Member type number
- Member type name
- Issue date
- Due date
- Return date
- Total fine
- Fine paid
- Balance

4.4 Output Design

Computer output is the most important and direct source of information to the user. User's need should be clean in the design of output. User can often be quite specific as to what he wants to use in front of them. Before designing the output some things are kept in view i.e. what output information is needed, how it should be presented, what format it should have, it is needed what the volume will be.

Outputs from a system can be in the form of queries and reports. In designing outputs, the following factors should be kept in mind.

- Outputs should be good looking.
- They should be easy to understand.
- Unnecessary information should be avoided.
- Purpose of output should be mentioned.

4.4.1 Screen Outputs

Since the output of the proposed system is online, output on the screen should be appropriate and effective. Screen output is necessary if data is needed on a screen for an endless variety of uses. The screen outputs of the proposed system take the forms of reports and screen query replies. There are three search forms for the purpose of searching a particular book.

- Search by title Form
- Search by author Form
- Search by subject Form

4.4.2 Reports

The complaint about computer reports is that they are too confusing to use. Usually management is accustomed to receive information from their assistants. Computer reports are excessively long and summary of data with few clues to help the management extract the needed information. Keeping in view the above-mentioned discrepancies, the reports of the proposed system were designed so they should be meaningful and informative. In all reports date, time and page number are placed where required. These reports along with the description and format are as below.

The convention used in all reports is N numeric, V varchar2 and Date stands for date data types.

- Acquisition of books report
- Monograph written by particular author report
- Detailed monograph information report
- Total number of books report
- Lost books report
- Damaged books report
- Overdue books report
- Available books in library report
- Number of books issued to a particular member report
- Books issued at some particular date report
- Total members of any particular type report
- Member's policy report
- Member's detail information report
- Members left the library report
- Acquisition of serials report

- Serials report
- Serials copies report

4.4.2.1 *Acquisition of books report*

The report will be used to have the record of all the books reaching library for selection. The format of the report is as below.

Predate	Monograph title	Author	ISBN	Publisher name	Source	Acquisition type
Date	V(50)	V(50)	V(20)	V(50)	V(35)	V(35)
Date	V(50)	V(50)	V(20)	V(50)	V(35)	V(35)
Total: N(10)						

Supplier name
Char(35)
Char(35)

4.4.2.2 *Monograph written by particular author report*

The report will be used to have the complete record of all the books written by particular author. The format of report is as below

Author
V(50)

Accession number	Subject	Title of monograph	Subtitle	Edition	Reference	Status	Location
N(10)	V(35)	V(50)	V(50)	N(10)	Char(1)	Char(1)	V(30)
N(10)	V(35)	V(50)	V(50)	N(10)	Char(1)	Char(1)	V(30)
Total:N(10)							

4.4.2.3 *Total number of books report*

The report shows the total number of books present in the library.

Accession number	Subject	Title of monograph	Subtitle	Edition	Author	Ddc number
N(10)	V(35)	V(50)	V(50)	N(10)	V(50)	V(20)
N(10)	V(35)	V(50)	V(50)	N(10)	V(50)	V(20)

Cutter number	Reference	Status	Location
V(3)	Char(1)	Char(1)	V(30)
V(3)	Char(1)	Char(1)	V(30)
Total:N(10)			

4.4.2.4 Lost books report

The report shows the books lost by the member or by any other means.

Accession number	Title	Member_id	Name of member	Lost date	Status	Replacement
N(10)	V(50)	N(10)	V(35)	Date	Char(6)	Char(7)
N(10)	V(50)	N(10)	V(35)	Date	Char(6)	Char(7)
Total:N(10)						

Book lost by	Amount	Concession
V(10)	N(10)	N(6)
V(10)	N(10)	N(6)

4.4.2.5 Overdue books report

The report contains the list of books that are overdue and fine after some days, automatically calculated.

Accession number	Title	Member_id	Name of member	Type number	Issue date
N(10)	V(50)	N(10)	V(35)	N(10)	Date
N(10)	V(50)	N(10)	V(35)	N(10)	Date
Total:N(10)					

Due date	Fine calculated
Date	N(10)
Date	N(10)

4.4.2.6 Available books in library report

The report shows the books available in the library at the present time. The format of the report is as below

Accession number	Subject	Title of monograph	Subtitle	Edition	Author	Ddc number
N(10)	V(35)	V(50)	V(50)	N(10)	V(50)	V(20)
N(10)	V(35)	V(50)	V(50)	N(10)	V(50)	V(20)
Total:N(10)						

Cutter number	Reference	Location
V(3)	Char(1)	V(30)
V(3)	Char(1)	V(30)

4.4.2.7 *Damaged Books Report*

The report is just like the lost book report; it is used to show the record of damaged books in the library.

Accession number	Title	Book damaged by	Member id	Name of member	Replacement
N(10)	V(50)	V(10)	N(10)	V(35)	Char(7)
N(10)	V(50)	V(10)	N(10)	V(35)	Char(7)
Total:N(10)					

Amount	Concession
N(10)	N(6)
N(10)	N(6)

4.4.2.8 Number of books issued to a particular member report

The report shows the total number books in possession of a particular member with different or the same issue date.

Member_id	
N(10)	
Name of member	Type of member
V(35)	V(15)

Accession number	Title of monograph	Issue date	Due date	Fine calculated
N(10)	V(50)	Date	Date	N(10)
N(10)	V(50)	Date		
Total:N(10)				

4.4.2.9 Books issued at some particular date report

The report shows the total books issued at some particular date given by the user. The format of the report is as below

Issue date
Date

Accession number	Title of monograph	Member_id	Name of member	Type number	Type name	Due date
N(10)	V(50)	N(10)	V(35)	N(10)	V(15)	Date
N(10)	V(50)	N(10)	V(35)	N(10)	V(15)	Date
Total:N(10)						

4.4.2.10 Total members of any particular type report

The report shows the members that belong to the particular member type category.

Type number	Type of member
N(10)	V(15)

Member_id	Name of member	Father's name	Address	Phone	Date of birth	Member status
N(10)	V(35)	V(35)	V(50)	V(15)	Date	Char(1)
N(10)	V(35)	V(35)	V(50)	V(15)	Date	Char(1)
Total:N(10)						

Expiry date	Gender	Security
Date	Char(1)	N(6)
Date	Char(1)	N(6)

4.4.2.11 Member's policy report

This report shows the specific category or type to which the member belongs along with the detailed policy of the member type such as the max books allowed to the member, security deposited, fine per day charged to him, etc.

Member_id	Name of member
N(10)	V(35)

Type number	Type name	Fine per day	Max books allowed	Maximum fine	Security	Days of circulation
N(10)	V(15)	N(3)	N(3)	N(6)	N(6)	N(3)

Days for fine validation
N(3)

4.4.2.12 Member's detail information report

The report shows the detail information about a particular member.

Member id	Type name
N(10)	V(15)

Member name	Father name
V(35)	V(35)

Address	Phone	Gender	Expiry date	Date of Birth	Member status	Security
V(50)	V(15)	Char(1)	Date	Date	Char(1)	N(6)

4.4.2.13 Members left the library report

The report specifies the persons who have cancelled their membership of library along with the details of security refund.

Member id	Member name	Type number	Type name	Leaving date	Security	Refund	Amount
N(10)	V(35)	N(10)	V(15)	Date	N(6)	Char(1)	N(6)
N(10)	V(35)	N(10)	V(15)		N(6)	Char(1)	N(6)
Total: N(10)							

4.4.2.14 Acquisition of serials report

The report specifies the serials reaching library for selection purpose. The report shows the serials marked as rejected by the authorities of library.

Predate	Title	ISSN	Publication type	Publisher name	Supplier name	Source
Date	V(50)	V(20)	V(50)	V(50)	Char(35)	V(35)
Date	V(50)	V(20)	V(50)	V(50)	Char(35)	V(35)
Total:N(10)						

Acquisition type	Rejected
V(35)	Char(1)
V(35)	Char(1)

4.4.2.15 Serials Report

The report shows the total number of serials along with their type reaching library.

Serial id	Language	Title	Subtitle	Subject	ISSN	Publication type	Frequency
N(10)	V(20)	V(50)	V(50)	V(40)	V(20)	V(20)	V(35)
N(10)	V(20)	V(50)	V(50)	V(40)	V(20)	V(20)	V(35)
Total:N(10)							

4.4.2.16 *Serial copies report*

The report shows the details of all serial copies of any particular serial.

Serial id	Title	Subtitle	Publication type	Frequency
N(10)	V(50)	V(50)	V(20)	V(35)

Serial number	Serial date from	Serial date to	Volume number	Acquisition type
N(10)	Date	Date	V(5)	V(35)
N(10)	Date	Date	V(5)	V(35)
Total: N(10)				

Source	Shelving location	Status
V(40)	V(25)	V(1)
V(40)	V(25)	V(1)

4.5 File Organization and Database Design

After designing the input and output, the designer begins to concentrate on file design or how data should be organized around user requirements. The entire system depends upon good file design. The problems of data duplication and data redundancy have been taken into account very carefully while developing the system design.

4.5.1 Database model

A Database is a shared collection of inter-related data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the user. And the data model is a method of representing, manipulating and retrieving information. [7] Hence in case of *General Library System* the database model adopted is “Relational Database Model” (RDMBS).

4.5.2 Data Structure

Data are structured according to the data model. The entity relationship diagram (ERD) of *General Library System* shows the data structured in a particular way. An entity is the conceptual representation of an object and the relationships between entities actually make up the data structure. [7]

4.5.3 Normalization

Data structuring is refined through a process called normalization. Data were grouped in the simplest way possible so that the later changes could be made with minimum of impact on the data structure. In case of *General Library System* normalization up to three levels have been made.

First Normal Form: For the first normal form, repeated groups have been removed, so there is single value at the intersection of each row and columns of the tables.

Second Normal Form: For the second normal form, partial functional dependencies have been removed.

Third Normal Form: For the third normal form, transitive dependencies have been removed.

4.5.4 File Design

On the basis of designing principles, the structure of the files along with the description is as below.

4.5.4.1 *AcqBooks*

Field Name	Type	Length	Null	Constraints	Description
Title	Varchar2	50	N	Primary Key	Records the main title of book
Author	Varchar2	50	N	Primary Key	Author name as composite key
Predate	Date		N		The date of delivery of book
Publisher_id	Number	10	N	Foreign Key	The reference of publisher is placed here, as there can be multiple publishers.
ISBN	Varchar2	20	Y		International standard books contain the ISBN.
Acqtype	Varchar2	35	N		Acquisition type means by which method the book has reached the library.
Source	Varchar2	35	Y		The source of supplying book.
Supplier_no	Number	10	N	Foreign Key	The reference of supplier, if the source is other than supplier then distinct number should be given to this category.
Rejected	Char	1	Y		The flag indicates whether book reached is under consideration or rejected

Description

The table is used to record the brief detail of the books reaching library for selection by different sources either by supplier or any other source. The primary key is concatenated key of “title” and “author” as both of them collectively making the unique criteria.

Primary Key(s): (Title+author)

Foreign Key(s): (publisher_id, supplier_no)

4.5.4.2 Bookorder

Field Name	Type	Length	Null	Constraints	Description
Order_no	Number	10	N	Primary key	Unique order number specified
Title	Varchar2	50	N	Foreign Key	Reference of acquisition book title as foreign key
Author	Varchar2	50	N	Foreign key	The concatenated foreign key as reference
Supplier	Number	10	N		Supplier number to whom the order has been placed
Orderreceive	Char	1	Y		The flag indicates that the order has reached the library or not
Invoice_no	Number	10	Y		The invoice number of the order reaching library
No_of_copies	Number	10	N		Total number of copies ordered
Iprice	Number	7	Y		The individual price of the book
Payment	Number	10	Y		Total payment made against order

Description

The table shows the books ordered for purchase after selection. After receiving the order, the details of payment are also entered in the table.

Primary Key(s): order_no
Foreign Key(s): (title+author)

4.5.4.3 Books

Field Name	Type	Length	Null	Constraints	Description
Book_id	Number	10	N	Primary key	The unique primary key that is actually the Accession number of the book in library.
Publisher_id	Number	10	N	Foreign key	The reference of publisher is placed here
Subject	Varchar2	35	N		The subject of the monograph
Title	Varchar2	50	N		Records the main title of the monograph
Subtitle	Varchar2	50	Y		As most of monographs have subtitles along with their titles.
Pub_date	Date		Y		Specifies the publication date of the monograph
Place	Varchar2	30	Y		The place of publication
Edition	Number	10	N		Edition number of monograph
Pages	Number	20	Y		Total number of pages
Author	Varchar2	50	N		Records the author name(s) of monographs
ISBN	Varchar2	20	Y		International standard book number
Reference	Char	1	N		The flag determines whether the monograph is for reference purpose or can be issued
Price	Number	10	Y		The individual price of each monograph
No_of_volumes	Number	10	Y		The volumes number of monograph
Status	Char	1	N		Specifies the current status of monograph ('P' for present, 'L' if lost and 'D' if damage)
Ddc_no	Varchar2	20	Y		The standard Dewey Decimal Classification number used for cataloguing
Cutterno	Varchar2	3	N		Composed of first three characters of author's last name
Supplier_no	Number	10	N	Foreign key	The reference of supplier who supplies the book
Notes	Varchar2	50	Y		Remarks
Location	Varchar2	30	Y		The shelving location in the library

Description

The table contains the descriptive information about the particular monograph in the library. A unique accession number is allocated to each copy of monograph.

Primary Key(s): (book_id)
Foreign Key(s): (supplier_no,publisher_id)

4.5.4.4 Supplier

Field Name	Type	Length	Null	Constraints	Description
Supplier_no	Number	10	N	Primary Key	Uniquely determines the supplier number
Sup_name	Char	35	N		Supplier's name
Address	Varchar2	50	N		Address of supplier
Phone1	Varchar2	15	Y		Phone number of supplier
Phone2	Varchar2	15	Y		Additional phone number
Email	Varchar2	25	Y		Email address if any
Web	Varchar2	25	Y		Any web URL or website address

Description

The table contains the information of suppliers in order to make it easy to contact with them.

Primary Key(s): (supplier_no)

4.5.4.5 Publisher

Field Name	Type	Length	Null	Constraints	Description
Publisher_id	Number	10	N	Primary Key	Uniquely determines publisher number
Name	Varchar2	50	N		Contains the name of publisher or publishing company
Address	Varchar2	50	N		Specifies the address of publisher
Email	Varchar2	25	Y		Email address in there is any
Phone	Varchar2	25	Y		Phone number of publisher
Fax	Varchar2	25	Y		Fax number
Web	Varchar2	40	Y		Any website URL

Description

The table contains the information of the publisher or the publishing company.

Primary Key(s): (publisher_id)

4.5.4.6 *Member_type*

Field Name	Type	Length	Null	Constraints	Description
Type_no	Number	5	N	Primary Key	Unique number assigned to each type
Type_name	Varchar2	15	N		Name of the type
Fine_per_day	Number	3	N		The fine charged per day when book is issued
Days_of_circulation	Number	3	N		Determines the number of days the monograph can be kept issued
Max_books	Number	3	N		Determines the number of books a member of this type can contain
Max_fine	Number	6	Y		Specifies the maximum fine after which no book will be issued to the member
Security	Number	6	Y		The security settled for the specific member type.
Days_for_finevalid	Number	3	N		Determines the days after which the fine per day starts affecting

Description

The table contains the privileges of all member types of the library on the basis of which circulation desk processing has been made.

Primary Key(s): (type_no)

4.5.4.7 Member

Field Name	Type	Length	Null	Constraints	Description
Member_id	Number	10	N	Primary Key	The uniquely assigned number to every member of library
Type_no	Number	5	N	Foreign Key	Specifies the type of member that determines the rights of member
Name	Varchar2	35	N		Contains the name
F_name	Varchar2	35	N		Specifies the father's name of member
Address	Varchar2	50	N		Specifies address of the member
Phone	Varchar2	15	Y		Records the phone number
Email	Varchar2	25	Y		The email address of member
D_O_B	Date		N		Date of birth of member
Security	Number	6	Y		Security specified by the particular member type
Concession	Number	6	Y		Any concession made to the member
Gender	Char	1	N		Specifies the gender, 'M' for male and 'F' for female
Memstatus	Char	1	N		Specifies that whether person is currently member of library or has been left the membership
Expdate	Date		Y		Determines the expiry date of membership
Remarks	Varchar2	40	Y		Remarks

Description

The table contains the descriptive information of all the members of the library.

Primary Key(s): (member_id)

Foreign Key(s): (type_no)

4.5.4.8 *Leaving_member*

Field Name	Type	Length	Null	Constraints	Description
Member_id	Number	10	N	Primary Key	Reference of member id from the member table
				Foreign Key	
Type_no	Number	5	N	Primary Key	Reference of type_no from member type
				Foreign Key	
Leaving_date	Date		N		The date of entry for which membership is cancelled
Recovery	Number	6	Y		The amount of security deposited by member at the time of membership
Refund	Char	1	N		The flag determines whether the security has been refunded or not
Amount	Number	6	Y		Specifies the balance amount refunded to the member
Reason	Varchar2	40			The reason of canceling membership

Description

The table shows the record of members whose membership has been cancelled by the library.

Primary Key(s): (member_id+type_no)
Foreign Key(s): (member_id,type_no)

4.5.4.9 Circulation

Field Name	Type	Length	Null	Constraints	Description
Member_id	Number	10	N	Primary Key	The borrower number of member who issues the monograph
				Foreign Key	
Type_no	Number	5	N		The type number assigned to the member for applying policy rules to the circulation processing
Book_id	Number	10	N	Primary Key	The unique accession number of the book that has been issued
				Foreign Key	
Issue_date	Date		N	Primary Key	The date at which the book was being issued
Due_date	Date		N		The date at which the book is due
Return_date	Date		Y		The date at which book is returned
Fine	Number	6	Y		Contains the amount of fine if member returns book after due date
Paid	Number	6	Y		The payment made by the member

Description

The table contains the transactions of circulation i.e. book issue and return.

Primary Key(s): (member_id+book_id+issue_date)

Foreign Key(s): (member_id, book_id)

4.5.4.10 *Lost_book*

Field Name	Type	Length	Null	Constraints	Description
Book_id	Number	10	N	Primary Key	The accession number of the book that is being lost / damaged
				Foreign Key	
Status	Char	6	N		The field shows that the book is lost or damaged
Lostby	Varchar2	10	N		The field determines whether book is lost by member or by some other means
Member_id	Number	10	Y	Foreign Key	The borrower number who issued the book
Lost_date	Date		Y		The date at which book is lost/ damaged
Replacement	Char	7	N		Field shows that in case of lost it has been replaced or not and in case of damage what method is adopted for the replacement
Amount	Number	10	Y		The fine or charges made in case of book lost / damage
Concession	Number	6	Y		Any concession made in the payment
Remarks	Varchar2	40	Y		Remarks

Description

The table is used to keep the record of books that are being lost or damaged by members or by other means.

Primary Key(s): (book_id)
Foreign Key(s): (book_id, member_id)

4.5.4.11 Serial

Field Name	Type	Length	Null	Constraints	Description
Serial_id	Number	10	N	Primary Key	The unique number assigned to serial. However it will be same for each copy of serial.
Publisher_id	Number	10	N	Foreign Key	The reference of publisher id
ISSN	Varchar2	20	Y		The international serial number
Title	Varchar2	50	N	Primary Key	Records the main title of serial
Subtitle	Varchar2	50	Y		Records the subtitle if there is any
Subject	Varchar2	40	Y		The subject related to the serial
Entry_date	Date		N		The date at which the entry is made
Lang	Varchar2	20	N		The language in which serial is published
Publicationtype	Varchar2	20	N		Specifies that the serial is a magazine, journal, documents, or any other sequential technical writing
Frequency	Varchar2	35	N		Specifies that the serial is weekly, monthly, or annually etc.
Subsdate	Date		Y		If there is any subscription date of serial
Expirydate	Date		Y		The expiry date of subscription
Renewaldate	Date		Y		The renewal date of subscription if the serial is subscribed again

Description

The table contains the descriptive information of a serial.

Primary Key(s): (serial_id+title)
Foreign Key(s): (publisher_id)

4.5.4.12 Serial_copies

Field Name	Type	Length	Null	Constraints	Description
Serial_no	Number	10	N	Primary Key	This is the unique number assigned to each copy of the serial
Serial_id	Number	10	N	Foreign Key	The field identifies the serial's attribute from the serial table
Serialdatefrom	Date		Y		The publication date of copy is recorded
Serialdateto	Date		Y		Depending upon the frequency of publication, the date till the next copy publishes is recorded
Title	Varchar2	50	N	Foreign Key	The reference of title from the serial table
Supplier_no	Number	10	N	Foreign Key	The reference of supplier who supplies the copies
Volume_no	Number	5	Y		Records the volume number of serial if there is any
Acqtype	Varchar2	35	N	Foreign Key	Records the reference of type of acquisition means by which means the serial reaches the library either by purchase or by any other means
Status	Varchar2	1	N		Status of serial ('P' for present, 'L for lost)
Source	Varchar2	40	N		Specifies the source of supplying serial
Shelvingloc	Varchar2	25	Y		The shelving location of serial in library
Remarks	Varchar2	40	Y		The remarks about serial

Description

The table contains the descriptive information of the copies of serials whose information is already stored in the 'Serial' table.

Primary Key(s): (serial_no)

Foreign Key(s): (serial_id+title, supplier_no, acqtype)

4.5.4.13 *Acqcopy*

Field Name	Type	Length	Null	Constraints	Description
Acqtype	Varchar2	35	N	Primary Key	Records the type of acquisition means by which means the serial reaches the library either by purchase or by any other means
Purchase_price	Varchar2	25	Y		In case of purchase acquisition type, this field is entered
Invoice_no	Number	15	Y		The field specifies the invoice number in case of purchase
Notes	Varchar2	40			Remarks

Description

The table is the relational table resulting from the resolution of transitive dependency in serial_copies table.

Primary Key(s): (acqtype)

4.5.4.14 *Serial_publisher*

Field Name	Type	Length	Null	Constraints	Description
Serial_id	Number	10	N	Primary Key	The number refers the serial in the serial table
				Foreign Key	
Publisher_id	Number	10	N	Primary Key	The number refers the publisher in the publisher table
				Foreign Key	

Description

The table stores the information of the publisher a particular serial has. It is basically a relational table, which relates 'Serials' & 'Publisher'.

Primary Key(s): (serial_id+publisher_id)
Foreign Key(s): (serial_id,publisher_id)

4.5.4.15 Acqserials

Field Name	Type	Length	Null	Constraints	Description
Title	Varchar2	50	N	Primary Key	Records the main title of the serial
Publisher_id	Number	10	N	Primary Key	As there can be different publishers that's why the reference is placed
				Foreign Key	
Predate	Date		N		The date of delivery of serial
Author	Varchar2	50	Y		The author of serial is recorded here
ISSN	Varchar2	20	Y		International standard serial number
Subtitle	Varchar2	50	Y		The subtitle if there is any
Publicationtype	Varchar2	50	Y		Specifies that the serial is a magazine, journal, documents, or any other sequential technical writing
Acqtype	Varchar2	35	N		The type of acquisition, either purchased, subscribed, free/ gift etc.
Source	Varchar2	35	N		The source of providing serial to library
Supplier_no	Number	10	N	Foreign Key	The reference of supplier number from the supplier
Rejected	Char	1	N		The flag determines whether the serial purchase is selected or rejected.

Description

The table is used to keep the record of serials coming to library by any means for selection purpose.

Primary Key(s): (title+publisher_id)

Foreign Key(s): (publisher_id, supplier_no)

4.5.4.16 Serialorder

Field Name	Type	Length	Null	Constraints	Description
Order	Number	10	N	Primary key	Unique order number specified
Title	Varchar2	50	N	Foreign Key	Reference of acquisition book title as foreign key
Supplier_no	Number	10	N	Foreign Key	Supplier number to whom the order has been placed
Publisher_id	Number	10	N	Foreign Key	The concatenated foreign key placed for the reference
Copies	Number	10	Y		Total number of copies ordered
Received	Char	1	Y		The flag indicates that the order has reached the library or not
Recievingdate	Date		N		The date for which order has reached the library
Invoiceno	Number	10	Y		The invoice number of the order reaching library
Payment	Number	10	Y		Total payment made against order

Description:

The table is used to keep the record of orders placing to the supplier for purchase.

Primary Key(s): (order)

Foreign Key(s): (publisher_id+title, supplier_no)

CHAPTER 5

IMPLEMENTATION

5. IMPLEMENTATION

The next phase of software lifecycle, after designing is system development and implementation. The software is developed to fulfill the proposed and design specifications. During this phase of development, design code or programs are written to meet all the desired requirements. This phase require specific plan for different programming techniques. After writing different codes and functions, their interfacing is required in order to achieve the target. The phase depends internally on the package or the programming language that we are using.

5.1 Software Tools Used

The system designed, as described in the previous ‘System Design’ chapter, is basically a DBMS application that requires proper database development. For making the system usable and functional, it is developed as GUI product. For data input and viewing records that have been saved as database file, Forms have been used and for the output in printed form, reports have been used. The tools used for implementing Database Design is ‘*Oracle 8.05*’, for Form Design is ‘**Form Developer6i**’ and for implementing Report Design is ‘**Report Developer6i**’.

5.1.1 Oracle 8.05 as Database engine

The choice of software is very important for the users point of view. It should be according to the users point of view so that the software would be user friendly. After considering the requirements and the need of today, “**ORACLE 8.05**” is proposed as the database engine. The reasons for selecting Oracle are as follows.

- Oracle offers a full implementation of SQL language, which is the standard relational database language.
- Oracle supports a multi-user system, which allows users to share data thus increasing throughput.
- Oracle can run on wide variety of machines.
- Oracle is flexible enough to run under PCs like DOS, VAX/VMS, UNIX and IBM VM/CMS. To offer an identical package for each of these platforms is not an easy job, but it is the specialty of oracle.
- Oracle controlled access to the databases. It protects data from unauthorized access and system failure.
- Oracle provides many component suites for building comprehensive applications with graphical interfaces (GUI).

5.1.2 Form Developer6i

Forms Developer is Oracle's productive Rapid Application Development (RAD) environment for building highly scalable, enterprise-class Internet database applications. It uses powerful declarative features so that business developers can instantly create fully functional applications from database definitions. The features that induced me to use Form Developer as front-end developing tool are:

- Deployment of applications on the Internet.
- Form Developer produce efficient, scalable applications.
- It builds large, consistent, maintainable applications quickly.
- Form Developer provides a set of integrated builders that enable a developer to construct fully functional database applications with minimal effort.
- The main builder in Forms Developer is Form Builder that is a powerful development tool for building robust, enterprise-class applications that enable end users to retrieve, enter, modify, and save information in the database.
- Using Form builder third party tools and database servers can be used.

5.1.3 Report Developer*6i*

Reports Developer is a powerful enterprise reporting tool used to build reports that dynamically retrieve, format, and distribute information stored in the database. Due to the following reasons, this tool has been used for report development:

- Report Developer*6i* provides the facility of web publishing high quality reports.
- It performs unlimited data formatting.
- Report Developer enables the developer to take advantage of application server based reporting.
- Integrated business intelligence solutions are provided to meet the user needs.
- The software in Report Developer provides a set of integrated builders that enable us to construct fully functional database applications with minimal effort.
- The main builder in Report Developer is Report Builder with all the design-time, compile-time, and runtime components.
- Report Developer includes additional tools to automate and manage application development tasks. These are Graphics builder, and Translation builder. Report Server component is also included that is used to deploy the application to the web.

5.2 Form Designing

The form-based applications of the system are developed in Form builder. Different forms have been designed that use a number of data sources, including Oracle databases. The highest-level Form Builder objects are *modules*. Each module consists of any number of lower-level objects. In the Form Builder environment, working with four types of modules, a complete application has been created. [8] Implementation has been

performed keeping in view the design of forms as described in previous chapter 'System Design'. The forms are designed and implemented with two types of modes.

- Data entry mode
- Modify mode

5.2.1 Data Entry Forms

The data entry forms for publisher, supplier, books, member, member type and lost book have the same method and criteria for implementation. Serials entry form has option for entering new serials as well as the copies of serials reaching library periodically. Leaving member entry form has the facility of viewing the list of books already issued to the member whose record is going to be inserted. Acquisition forms have both the data entry and view mode with order placing facility. Book issue form has lots of checks and validation for correct entry of data. Using Form Builder form modules of data entry forms have following methodology.

The main form modules implemented for data entry forms are different levels triggers, PL/SQL libraries, and a menu module. Form module has further some objects having codes and features.

5.2.1.1 Form level Triggers

Triggers are PL/SQL blocks executed on an event. Form level triggers used for publisher, supplier, books, members, member types, and lost book forms are as follows.

- ***When-New-Form-Instance***

At form start-up, Form Builder navigates to the first navigable item in the first navigable block. A When-New-Form-Instance trigger fires after the successful completion of any navigational triggers that fire during the initial navigation sequence. The function for maximizing form size is written as PL/SQL code.

```
Set_window_property(forms_mdi_window,window_state,maximize);
Set_window_property('window1',window_state,maximize);
```

- **On-Message**

Fires whenever Form Builder would normally cause a message to display and pre-empts the message. The form builder messages are overridden by the messages written in PL/SQL code. The PL/SQL code written for the trigger is [9]

```
DECLARE
    msgnum NUMBER      := MESSAGE_CODE;
    msgtxt VARCHAR2(80) := MESSAGE_TEXT;
    msgtyp VARCHAR2(3)  := MESSAGE_TYPE;
BEGIN
    IF msgnum = 40400 THEN
        Message('Your changes have been made permanent.');
    ELSIF msgnum = 40401 THEN
        Message('You have no unsaved changes outstanding.');
    ELSE
        /*  ** Print the Normal Message that would have appeared
        **  ** FRM-12345: Message Text Goes Here  */
        Message(msgtyp||'-'||TO_CHAR(msgnum)||': '||msgtxt);
    END IF;
END;
```

- **On-Error**

An On-Error trigger fires whenever Form Builder would normally cause an error message to display. The PL/SQL code written in the trigger is as follows:

```

DECLARE
    msgnum NUMBER      := MESSAGE_CODE;
    msgtxt VARCHAR2(80) := MESSAGE_TEXT;
    msgtyp VARCHAR2(3) := MESSAGE_TYPE;
BEGIN
    IF msgnum = 40200 THEN
        Message('You cannot change the record');
    ELSIF msgnum = 40202 THEN
        Message('The field cannot be left null');
    ELSIF msgnum = 50016 THEN
        Message('Improper format, enter the correct one');

    ELSE
        /* ** Print the Normal Message that would have appeared
        ** ** FRM-12345: Message Text Goes Here */
        Message(msgtyp||'-'||TO_CHAR(msgnum)||': '||msgtxt);
    END IF;
end;

```

- **When-Window-Closed**

Fires when an operator closes a window using a window-manager specific Close command. The system has been designed to transfer the control back to the main screen whenever the window is closed.

- **Key_Down**

Fires when the user attempts to press the down arrow key. This trigger has been used to avoid inserting next record while pressing the arrow keys. The code written is

```

if :system.last_record <> 'TRUE' then
    next_record;
end if;

```

5.2.1.2 Canvases

A canvas is the background object upon which interface items and graphic elements appear. Canvases are of four types namely content, stacked, tab and toolbar. Single canvas has been created in data entry forms.

5.2.1.3 Alerts

Alerts are the modal windows that display a message. In the development of system alerts have been used to inform the user of unusual situations or to give a warning for an action that is undesirable. The function used for displaying alert is SHOW_ALERT(). This function returns alert_button1, alert_button2 or alert_button3 based on the button clicked by the user. [9] An alert named 'Confirm_alert' is called when the user wants to exit any particular form. The code for calling alert is:

```
if :system.form_status = 'CHANGED' then
    set_alert_property('confirm_alert',title,'Save Changes');
    set_alert_property('confirm_alert',alert_message_text,'Would you
like to make changes permanent');
    answer := show_alert('confirm_alert');
    if answer = ALERT_BUTTON1 then
        commit_form;
    end if;
```

5.2.1.4 Generation of Sequence number

For every data entry form, a unique sequence number has been generated every time when the user inserts a new record. '*When Create Record*' trigger has been used at block level to generate the sequence number.

5.2.1.5 Data Blocks and Triggers used

Two data blocks have been created for the data entry forms of Publisher, Supplier, Lost book, Books, Member, and Member type forms. One data block is for the database table and other is Control block containing the buttons. The data blocks depending upon tables contain the text fields, check boxes, list items, and radio groups for data entry along with the triggers associated with them.

5.2.1.5.1 *Block Level Triggers*

The triggers at the block level are '*Key_Up*', '*Key_Down*', and '*When_Create_Record*' for every data entry form along with some more triggers like '*Key_Delrec*' to avoid deleting records in Publisher, Supplier, and Books forms. '*When_New_Record_Instance*' for controlling radio groups in Lost book and Members forms. '*Post_Insert*' for modifying the status of book that has been considered lost in Lost Books form.

5.2.1.5.2 *Buttons*

The control blocks have Buttons that are used to call the event related to it. Some buttons created for data entry forms are common in use. These buttons are provided with shortcut keys.

- Add/Save for adding the new record by saving the existing one.
- Exit button for exiting the form.
- Some more buttons for calling List of Values according to the need of form.

For adding functionality to the buttons '*When_Button_Pressed*' trigger has been used with some PL/SQL code.

5.2.1.5.3 *Item Level Triggers*

At the level of text items '*When_Validate_Item*' trigger has been used in every form for checking the consistency and validity of data entered. '*When_Radio_Changed*' trigger has been used to control the radio buttons in Member, Books, and Lost books forms.

5.2.1.6 List of Values and Record groups

With every data entry form, some lists of values (LOVs) have been developed for providing the end user either a single or multi-column selection list. List of values for Publishers, Suppliers, Ordered books, Members list, and Issued books in case of lost book have been created in different forms

5.2.1.7 Property Classes

A property class is a named object that contains the list of properties and their values. [8] The property class PCLASS_BUTTON has been created for the buttons so that all the buttons based on the class inherit the settings of the properties in the class.

5.2.1.8 Attached Libraries

These are read-only files. An attached library is not editable in the form or menu module. These are the PL/SQL libraries that are created, compiled and saved to the file system. The library attached to the data entry form modules is ENB.dll that has two procedures.

5.2.1.9 Menu Module

The menu named menu.mmb has been created with every form. The objects of menu are Action, Sources, Members, Books, Serials, Circulation and Backup. Each menu contains submenus of related forms and reports that have the PL/SQL source code for calling the forms or reports from the menu list.

5.2.1.10 PL/SQL Libraries

PL/SQL library is composed of program units and attached libraries. The system contains a PL/SQL library named ENB.PLL. The library ENB.PLL has two procedures named as Check_Record_Position and Item_Enable_Disable. The code written for these procedures is as follows. [9]

```

PROCEDURE check_record_position IS
BEGIN
  IF name_in('System.Cursor_Record') = '1' AND name_in('System.Last_Record') =
'TRUE' THEN
    Set_Item_Property('control.cmd_first',enabled, property_off);
    Set_Item_Property('control.cmd_previous',enabled, property_off);
    Set_Item_Property('control.cmd_next',enabled, property_off);
    Set_Item_Property('control.cmd_last',enabled, property_off);
  ELSIF name_in('System.Cursor_Record') = '1' THEN
    Set_Item_Property('control.cmd_first',enabled, property_off);
    Set_Item_Property('control.cmd_previous',enabled, property_off);
    Set_Item_Property('control.cmd_next',enabled, property_on);
    Set_Item_Property('control.cmd_last',enabled, property_on);
  ELSIF name_in('System.Last_Record') = 'TRUE' THEN
    Set_Item_Property('control.cmd_first',enabled, property_on);
    Set_Item_Property('control.cmd_previous',enabled, property_on);
    Set_Item_Property('control.cmd_next',enabled, property_off);
    Set_Item_Property('control.cmd_last',enabled, property_off);
  ELSE
    Set_Item_Property('control.cmd_first',enabled, property_on);
    Set_Item_Property('control.cmd_previous',enabled, property_on);
    Set_Item_Property('control.cmd_next',enabled, property_on);
    Set_Item_Property('control.cmd_last',enabled, property_on);
  END IF;
END;

```

```
PROCEDURE item_enable_disable1(blk_name IN char,item_on_off IN NUMBER) IS
    nxt_itemname varchar2(70);
    itemtype varchar2(25);
    itemcanvas varchar2(25);

BEGIN

/* get_block_property is a function that gets the value of specific block property.
the First_item_property holds the name of first enterable item in block. */

    nxt_itemname := get_block_property(blk_name,first_item);

    loop

        itemtype := get_item_property(blk_name||".||"||nxt_itemname,item_type);
        itemcanvas := get_item_property(blk_name||".||"||nxt_itemname,item_canvas);

        if itemtype <> 'DISPLAY ITEM' and itemcanvas is not null then

            set_item_property(blk_name||".||"||nxt_itemname,update_allowed,item_on_off);
            end if;

            nxt_itemname := get_item_property(blk_name||".||"||nxt_itemname,NEXTITEM);
            if (nxt_itemname = get_block_property(blk_name,last_item)) then
                exit;
            end if;
        end loop;

END;
```

5.2.2 Leaving Member Form

The form for the members leaving the membership has some features and objects as present in Data entry forms like canvases, attached libraries, PL/SQL libraries, Form level triggers, Property classes, Menu module, and Attached library. The difference is with the Data blocks. The form has three data blocks, Leaving member data block, Control block and the Circulation data block having the list of books already issued to the member whose record is being inserted as leaving member. The leaving member data block has a field of Member_id that is double clicked to display the books issued to the member entered in the field. The function code has been written in the trigger '*When_Mouse_Doubleclick*' with the function of fine calculation.

5.2.3 Acquisition Forms

The acquisition forms for Books and Serials have the same entry and view mode of implementation. The properties of Canvases, Alerts, Attached Libraries, Property classes are same as of Data Entry forms described above. The acquisition forms are basically Master detail forms with two data blocks having relation with each other. Due to the Master Detail nature, the Form Level triggers have an additional trigger named '*On_Clear_Details*'.

5.2.3.1 Data Blocks and triggers

There are three data blocks in acquisitions forms of books and serials. One is the Control Block with items Buttons and two blocks are based on Acquisition tables. The Master data block has Form layout style and Detail block has Tabular Layout style. The Master block has the relation with Detail block having triggers '*On_Populate_Details*' and '*On_Check_Delete_Master*'. The detail data block has 6 record places with a vertical scrollbar.

5.2.3.2 Program Units

The program units in Acquisition forms have three procedures created because of the master-detail nature of forms. These procedures are ‘Check_Package_Failure’, ‘Clear_All_Master_Details’, and ‘Query_Master_Details’.

5.2.4 Book Issue Form

The Book Issue form is also a type of data entry developed with some different features. The Canvases, Alerts, Menu module, and Form level triggers are the same as of Data entry forms.

5.2.4.1 Data Blocks and triggers

The book issue form has four data blocks with Master-Detail nature. The detail blocks have the relation with the master block having ‘Cascading ‘behavior. One Data block is in Tabular Layout model used to show the books already issued to the member whose data is going to be inserted in the field of Master Block. On double clicking the field, the user can get the books issued in the detail tabular block.

5.2.4.1.1 *Block level triggers*

Due to the Master-detail nature and cascading behavior the master data block have the triggers ‘On_Populate_Details’ and ‘Pre_Delete’ along with the other ‘Key_Up’ and ‘Key_Down’ triggers.

- *On_Populate_Details*

The PL/SQL code written in the trigger is as follows:

```

DECLARE
    recstat  VARCHAR2(20) := :System.record_status;
    startitm VARCHAR2(61) := :System.cursor_item;
    rel_id   Relation;
    --
    -- End default relation declare section
    -- Begin default relation program section
    BEGIN
        IF ( recstat = 'NEW' or recstat = 'INSERT' ) THEN
            RETURN;
        END IF;
        --
        -- Begin CIRCULATION1 detail program section
        --
        IF ( (:CIRCULATION.MEMBER_ID is not null) ) THEN
            rel_id := Find_Relation('CIRCULATION.CIRCULATION_CIRCULATION1');
            Query_Master_Details(rel_id, 'CIRCULATION1');
        END IF;
        --
        -- End CIRCULATION1 detail program section
        -- Begin MEMBER detail program section
        --
        IF ( (:CIRCULATION.MEMBER_ID is not null) ) THEN
            rel_id := Find_Relation('CIRCULATION.CIRCULATION_MEMBER');
            Query_Master_Details(rel_id, 'MEMBER');
        END IF;
        --
        -- End MEMBER detail program section
        -- Begin MEMBER_TYPE detail program section
        --

```

```

IF ( (:CIRCULATION.TYPE_NO is not null) ) THEN
    rel_id :=

    Find_Relation('CIRCULATION.CIRCULATION_MEMBER_TYPE');

    Query_Master_Details(rel_id, 'MEMBER_TYPE');

    END IF;

    --
    -- End MEMBER_TYPE detail program section

    IF ( :System.cursor_item <> startitm ) THEN
        Go_Item(startitm);
        Check_Package_Failure;
    END IF;

    END;

    --
    -- End default relation program section

```

- *Pre_Delete*

This trigger is fired because of the cascading behavior of the deleting record in Master detail form. The PL/SQL code written in the trigger is as follows.

```

-- Begin default relation program section

BEGIN

    -- Begin CIRCULATION1 detail program section

    DELETE FROM CIRCULATION C
    WHERE C.MEMBER_ID = :CIRCULATION.MEMBER_ID;
    -- End CIRCULATION1 detail program section

    -- Begin MEMBER detail program section

    DELETE FROM MEMBER M
    WHERE M.MEMBER_ID = :CIRCULATION.MEMBER_ID;

```

```

-- End MEMBER detail program section
-- Begin MEMBER_TYPE detail program section

DELETE FROM MEMBER_TYPE M
WHERE M.TYPE_NO = :CIRCULATION.TYPE_NO;
-- End MEMBER_TYPE detail program section
END;
-- End default relation program section

```

5.2.4.1.2 Items Functionality

The book issue form is very important as related to data manipulation. Various data validation checks have been applied to the fields of Accession number and Member id. The code has been written in the trigger '*When_Validate_Item*'. The checks have been applied to avoid invalid accession number, as the book is not present in the library, lost, damaged or issued to someone else. Same is the case with the Member id and Member type number. A trigger named '*When_Mouse_Doubleclicked*' with the Member id field has been used to display the books issued in the tabular data block.

To validate the Due date entered, '*When_Validate_Trigger*' has been used to the field. For displaying Due date automatically on double clicking the field, '*When_Mouse_Doubleclicked*' trigger has been used.

- *When_Validate_Item on Due Date*

```

BEGIN
--For checking the due date must be greater than issue date
  if :circulation.due_date < :circulation.issue_date THEN
    MESSAGE('Due date is before Issue date');
    RAISE form_trigger_failure;
  END if;
END;

```

- *When_Validate_Item on Accession number*

DECLARE

```

book circulation.Book_id%type;
lbook circulation.Book_id%type;
prebook books.book_id%type;

cursor primary_cur is
select book_id from circulation where return_date is null
and book_id = :CIRCULATION.BOOK_ID;

cursor status_cur is
select book_id from books where status = 'L'
and book_id = :circulation.book_id;

cursor second_cur is
select book_id from books
where book_id = :CIRCULATION.BOOK_ID;

```

BEGIN

```

-- for checking book's status
open status_cur;
fetch status_cur into lbook;
if status_cur%found then
  message ('Book lost');
  raise form_trigger_failure;
end if;
close status_cur;

-- for checking availability of the book in library

```

```

open primary_cur;
fetch primary_cur into book;
if primary_cur%found then
    message('Book already issued');
    raise form_trigger_failure;
end if;
close primary_cur;

--for checking invalid accession number entry
open second_cur;
fetch second_cur into prebook;
if second_cur%notfound then
    message('Invalid book entry');
    raise form_trigger_failure;
end if;
close second_cur;

END;

```

5.2.4.2 Program Units

The program unit module contains procedures. Some procedures are generated by the Form builder because of the Master Detail nature of the form as well as of the cascading behavior of the form. These procedures are '*Check_Package_Failure*', '*Check_All_Master_Details*', and '*Query_Master_Details*'.

5.2.4.3 LOVs and Record Groups

The list of values for Member id and Accession number of books have been used in order to select the values from a given list. The record groups contain the queries related to the list of values.

5.2.5 Book Return Form

This form is used for entering the books returned to the library. It has been developed as the view form with different options as of any other Data entry form. The similarity is only in the case of menu module, alerts, and property classes. The modules that have been implemented in the development of the form are as follows.

5.2.5.1 Canvases

Two canvases have been created in the form module. One canvas contains one data block with one text field and two buttons and it is displayed first when the form is instantiated. While the other canvas is the main canvas, that contains all the other major data blocks.

5.2.5.2 Screen Design

The form is developed in such a way that for entering the return date of a particular book, the user has to enter the accession number of the book to view its record. A button with label 'Search' is also provided with the form to search further records.

5.2.5.3 Data Blocks and triggers

Total six data blocks have been created for the form, one for the '*idcanvas*' and five for the '*maincanvas*'. The form is a Master-Detail form in which the three detail data blocks have relations with master data block having 'Cascading' behavior while one detail data block of Damage book has 'Non-Isolated' behavior. One Data block is in Tabular Layout model used to show the other books issued to the member whose record is displayed in the field of Master Block for return. This record has been displayed using a trigger '*Post_Query*' in the detail data block.

The Block level triggers are the same as in the previous Master Detail forms. With the Damage book data block a trigger has been created named '*Post_Insert*', to change the status of the book as Damaged in the *Lost_book* Table.

- *Post_Insert trigger on Data block*

The code written for the trigger is as follows.

```

BEGIN
  IF Form_Success THEN
    if checkbox_checked('lost_book.STATUS') then
      UPDATE BOOKS
      SET status = 'D'
      where book_id = :circulation.book_id;
    end if;
  end if;
END; [2]

```

5.2.5.4 Program Units

Along with the Master Details procedures of Form Builder, there are some other procedures such as 'Item_Enable_Disable' and 'Check_Record_Position' with the source code as described above.

5.2.5.5 Items Functionality

On entering the Return date, the system will automatically give the Calculated fine of overdue book depending upon the Type of member. For this calculation '*When_Validate_Item*' trigger has been used with the PL/SQL code written. The code is:

```

DECLARE
  days member_type.days_of_circulation%type;
  dfine member_type.fine_per_day%type;
  dvalid member_type.days_for_finevalid%type;
  v_days number(10);
  calcdtate date;

BEGIN
  IF :circulation.return_date < :circulation.issue_date THEN
    MESSAGE('Return date is less than issue date');
    RAISE form_trigger_failure;
  END IF;
  select member_type.days_of_circulation, member_type.fine_per_day,
  member_type.days_for_finevalid
  into days,dfine,dvalid
  from member_type
  where type_no= :circulation.type_no;

  --for calculating fine

  calcdtate := :circulation.due_date + dvalid;
  if :circulation.return_date < calcdtate then
    :circulation.fine := 0;
  else v_days := :circulation.return_date - calcdtate;
    :circulation.fine := dfine * v_days;
  end if;

END;

```

5.2.6 Modify Forms

The modify forms for publisher, supplier, books, member, member type, lost book and leaving member have the same method and criteria for implementation. Using Form Builder, form modules of modify forms have many modules in common with the other data entry forms as described above. Such as the Form level triggers are the same as described earlier in the Data entry forms.

The Modify form modules have been developed with some other objects having codes and features.

5.2.6.1 Canvases

Two canvases have been developed for the form module. One canvas contains one data block with one text field and two buttons and it is displayed first when the form is instantiated. While the other canvas is the main canvas, that contains all the other major data blocks.

5.2.6.2 Screen Design

The form is developed in such a way that every time the form is instantiated, user has to enter the id or some other unique value whose record has to be modified. A List of Value has also been created for selecting the record from list. A button with label 'Search' is also provided with the form to search further records.

5.2.6.3 Data blocks and Triggers

The modify forms of Publisher, Supplier, Books, Member, Lost books and Leaving Member have two data blocks while Member type form has an additional data block for displaying the list of members related to some particular member type. The forms of Books, Members, Lost Books, and Leaving member have a trigger in common that is '*Post_Query*' for displaying the fields not included in the data block table.

5.2.6.3.1 Block Level triggers

The triggers used for the two or three blocks are according to the needs of the system. Along with the triggers 'Key_Up', 'Key_Down' and 'Key_Delrec', nearly all the forms have 'Post_Query' trigger in the main data block. For example

- *Post_Query trigger on Books Data block*

DECLARE

```
cursor publish_cur is select pub_name
from publisher
where publisher_id = :books.publisher_id;
```

```
cursor supply_cur is select sup_name
from supplier
where :books.supplier_no IS NOT NULL
and supplier_no = :books.supplier_no;
```

BEGIN

```
open publish_cur;
fetch publish_cur into :books.cmd_pubname;
close publish_cur;
```

```
open supply_cur;
fetch supply_cur into :books.cmd_supname;
close supply_cur;
```

END;

5.2.6.3.2 Buttons

The buttons created for the modify forms are used to call the event related to it. Two buttons are on the '*mod_canvas*' and some more buttons on the main canvas that are common in use. These buttons are provided with shortcut keys.

- Show button on the '*mod_canvas*' to show the record of the id entered.
- Select from list on '*mod_canvas*' to call list of value.
- Delete Button for deleting the existing record if it is allowed to delete.
- Add/Save for adding the new record by saving the existing one.
- Save button for saving the records modified.
- Exit button for exiting the form.
- Search button for modifying more records.
- Print button for displaying the printed report.
- Some more buttons for calling List of Values according to the need of form.

For adding functionality to the buttons '*When_Button_Pressed*' trigger has been used with some PL/SQL code.

5.2.6.4 Program Units

Instead of using the attached PL/SQL library, I have created procedures of '*Check_Record_Position*' and '*Item_Enable_Disable*' as program units because all the items are not allowed to be modified. The items that cannot be modified are made disable.

5.2.6.5 LOVs and Record Groups

Along with the LOVs created in the data entry forms, the modify forms have one LOV additional. That is for selecting the record to be modified in the '*mod_canvas*'. The lists of values are multi record having all the records in the database table.

5.2.6.6 Property Classes

Along with the PCLASS_BUTTON property class for buttons, an additional class PCLASS_LOV for LOVs has also been created for inheriting the same features of list of values as all the modify forms have.

5.2.7 Serial Form

The serial form is also a Master-Detail form having relation with each other. This form has some features as present in the modify forms. Such as the property classes and menu module. It has also been created with two canvases.

5.2.7.1 Screen Design

The form is developed in such a way that every time the form is instantiated, user is provided the option for entering the record for a 'New serial' or adding the record of a 'New copy of an existing serial'. For this purpose a text box for entering the title of serial is created with a button 'Show' to view the serial entered previously. A List of Value has also been created for selecting the record from list. A button with label 'Search' is also provided with the form to search further records.

5.2.7.2 Data blocks and triggers

The data block nature of the form is different from the other Master-Detail forms. There is one data block for the 'idcanvas' and the other blocks named 'Serial', 'Serial_copies' and 'Acqcopy' have been created for the other main canvas. 'Serial' data block is master block for the 'Serial_copies' data block and 'Serial_copies' is master block for 'Acqcopy' data block. Control block contains the buttons.

- Add New Serial button for inserting new serial record.
- Save for saving the record entered.

- Add/Save for adding the new record by saving the existing one.
- Add a copy button for inserting a new copy of the serial.
- Exit button for exiting the form.
- Search button for searching a new record of serial.
- Some other buttons for calling the List of values.

5.2.8 Search Forms

There have been developed three search forms for searching a particular book either by entering title, author, or subject. The format of search forms is the same with a difference of PL/SQL code written. Two data blocks have been created in the forms, one data block having one text item and one Search button and the other data block to display the list of books for the particular search. Two more buttons have also been created, for printing the list and for exiting the form.

5.3 Report Designing

Report Builder is an important tool to design professional, production-quality reports. Various printed reports have been developed using the Report Builder development tool. A report developed is made up of objects, the building blocks that define the various parts of the report. Some objects, like fields, are visible in the report and others, like queries, work behind the scenes. Reports have been created keeping in view the two models that determine the look and feel of a report: [10]

- The data retrieved for a report has been defined in the **data model**.
- The style of a report has been defined in the **layout model**.

5.3.1 Development of report using Wizard

The reports have been developed the wizard provided by the ‘Report Builder’. The Report Wizard’s application-generation capabilities made me free to work on other aspects of the project reports. The wizard has guided step-by-step through the process of creating a report and I specified the options I wanted, and the Report Wizard did the rest, creating the appropriate data model and layout model, and displaying the report output automatically in the Live Previewer.

Wizards helped me to build reports and charts by asking for all the basic information needed in an easy-to-understand format. I have used some of the components as provided by the builder and some are customized according to the need of the system.

5.3.1.1 Style of Report

The reports have been given a style by selecting from the given style by the wizard. The reports of Available books, Acquisition books, Acquisition serials, Over due books, Serials, Books, Lost books, Damaged books and Members left has been developed with ‘Tabular’ style. The report for the Notices has been developed in the ‘Form Letter’ style while the rest of reports like Books issued to a particular member, Books issued at some date, Members of a particular member type, Member type policy report, and Serial copies report have been developed in ‘Group above’ style.

5.3.1.2 Query type and Data

After choosing the query type as SQL, the data for your report have been created by specifying a query. A query is a SQL statement that retrieves data from a standard datasource. I have written some queries directly in the space provided and for some other reports the visual Query Builder has been used for selecting the tables and data columns.

5.3.1.3 Selecting the fields

Fields correspond to the datasource columns selected by the query. By using the wizard I have selected all or just some of the fields to appear in the report's output.

5.3.1.4 Specifying Totals

Totals are computations performed on all of the records in a datasource column that have been specified by using the wizard of Report builder. This feature automatically creates the Formula and Summary columns in the Data Model.

5.3.1.5 Labels and Templates

The labels for the data fields have been customized and template of the report has been selected from the given list. Templates are pre-defined formats that have been applied to the reports to give them a consistent, professional look.

5.3.2 Editing in Live Previewer

The Live Previewer is a WYSIWYG editing environment for refining the appearance of the report. [10] Working directly with report output in the Live Previewer, I inserted the page number by clicking the Insert Page Number toolbar button, edited the text objects, changed the colors of objects, by selecting the object and choosing a new color from the color palette, resized the columns by dragging one of its select markers either horizontally or vertically to the new position and insert a field, click the Field tool and then click and drag an area in the Live Previewer.

5.3.3 Creating Data Model

The data model is composed of objects that define the data to be included in a report. It has been created for every report with some different objects. Data model objects included:

5.3.3.1 Queries

I have created the Queries by selecting the data from backend datasource Oracle using SQL SELECT statements. The option for taking input from parameter form has also been written in the queries. Such as for the report Members related to some particular type needs an input of Member type from the Parameter form to show the list of members. The Query written for the report is as follows.

```
SELECT ALL MEMBER_TYPE.TYPE_NO, MEMBER_TYPE.TYPE_NAME,  
MEMBER_MEMBER_ID, MEMBER_NAME, MEMBER_F_NAME,  
MEMBER_SECURITY, MEMBER_GENDER, MEMBER_ADDRESS,  
MEMBER_PHONE, MEMBER_D_O_B, MEMBER_EXPDATE,  
MEMBER_MEMSTATUS  
FROM MEMBER_TYPE, MEMBER  
WHERE MEMBER_TYPE.TYPE_NAME = :type  
AND (MEMBER_TYPE_NO = MEMBER_TYPE.TYPE_NO)
```

Here 'type' is the input from parameter form.

5.3.3.2 Columns

Columns represent the columns in the datasource that have been selected by the query. Some columns have also been defined that are based on computed values such as summaries (also called totals) and values set conditionally.

5.3.3.3 Groups

Groups organize the data into sets and hierarchies. By default, one group has been created for each query in the report. The group contains all the columns selected by the query.

5.3.3.4 User Parameters

User parameters are the variables that have been defined while developing some reports to assign the values for runtime. The code for assigning the parameter has been written in the queries of the report data model. The reports that contain the user parameters are:

- Book issued to the member report with Accession number as parameter
- Total books issued at a particular date with Issued date as parameter
- Total books in possession of a member report with Member id as parameter
- Total members related to a type report with Member type as parameter
- Lost books report with status and recovery as parameters
- Damaged books report with status and recovery as parameters
- Member Policy report with Member id as parameter
- Serial copies report with Serial id as parameter
- Notices with Accession number of book as parameter

5.3.4 Program Units

The reports that are related with the circulation desk processing that is Book issued to the member report, Total books issued at a particular date, Overdue books report, and Total books in possession of a member report and Notices report has a function for calculating fine with a formula column in the data model. This function returns a value to the formula column in the report. The function body contains the PL/SQL code as follows:

```
function tfineFormula return Number is

    days member_type.days_of_circulation%type;
    dfine member_type.fine_per_day%type;
    ddate circulation.due_date%type;
    dvalid member_type.days_for_finevalid%type;
    v_days number(10);
    calcdate date;
    tfine number(10);

begin
    --selecting member type for calculation of fine

    select member_type.days_of_circulation, member_type.fine_per_day,
    member_type.days_for_finevalid
    into days, dfine,dvalid
    from member_type
    where type_no= :type_no;

    select circulation.due_date into ddate
    from circulation
    where book_id = :book_id
    AND return_date IS NULL;

    --calculating fine as compared with the current system date

    calcdate := ddate + dvalid;
    if calcdate < sysdate then
        v_days := sysdate - calcdate;
        tfine := dfine * v_days;
```

```
else
    tfine := 0;
end if;
return tfine;
end;
```

5.4 Hardware and Operating System Requirement

The minimum hardware and operating system selection specification for running the Oracle 8.05, Form Developer*6i* and Report Developer*6i* is as below:

Main Processor	Pentium II
Main Memory	64 MB
Hard Disk	2.0 GB
Operating System	Windows NT/Win2000/winXP

Therefore depending upon the tools and system requirements following hardware has been used for the project implementation:

Main Processor	Pentium 4
Main Memory	128 MB
Hard Disk	40 GB
Operating System	Windows XP

CHAPTER 6

TESTING

6. TESTING

Software testing is a critical element of Software Quality Assurance and represents the ultimate review of specification, design, and coding. In this chapter, I have discussed the software testing fundamentals and the technique used for testing the product. Software testing fundamentals define the overriding objectives for software testing.

6.1 Testing Objectives

During testing, the program to be tested is executed with a set of test cases, and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected to. Keeping in view the following objectives in mind, the system has been tested. [6]

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an as-yet undiscovered error.
- A successful test is one that uncovers an as-yet undiscovered error.
- Tests are planned as soon as the requirements model is completed.
- A good test should be neither too simple nor too complex.

6.2 Testing Criteria

System should be developed to meet the objectives of proposed system as described earlier. The system has been tested in the light of some principles and a criterion is specified for the process. Testing of software is performed through various

steps, by entering records in data entry forms, by retrieving data in modify forms, by navigating through main screen to every form. The principles used for testing are as follows:

6.2.1 Operability

The system is tested in order to develop it into easily operable system. GUI interfaces provide easy and quick access to the system. Menus are provided for opening other forms or reports from anywhere in the system. Buttons are there for the user for insert, delete, save, modify, search and print the records that make the system much easy to operate. List of values are provided for the selection of any particular record from list.

6.2.2 Usability

Another objective of the system is usability. The system should be a usable product that's why it is tested in order to meet the requirements of usability. For this purpose various hints and tooltips are provided to the user. Guide messages are there for providing ease. Interface is designed for quick functioning of system. User needs not to waste time while finding buttons, shortcut keys, events related with the fields. Forms builder provides runtime messages and alerts for the user.

6.3 Testing Menu

The '*Main Screen*' is equipped with a menu to provide access to all the forms and reports of the system. The main menu has the submenus having code to access the forms and reports of the system. For testing the performance of menu, each form is called from the main screen. The reports have been called to check the performance of reports. On exiting the form, the control comes back to the main screen and it is displayed every time when the user exits any form.

6.4 Testing Data Entry forms

The data entry forms have been called from the menu provided at the main screen. The book entry form has been called to enter the books record. The form is opened with a sequence number for Accession number of book generated by the system. ‘*Publisher id*’ and ‘*Supplier number*’ has a Button for calling List of value. The publisher id has been selected from the list hence entering in the field. If the source is the Supplier then the supplier id is selected from the supplier id. The title field has been left empty for checking the validity check. The error message ‘*The field could not be null*’ was displayed. All the entries have been entered and the button ‘*Add/Save*’ has been pressed to save the entries placed and to make the system ready for the next entry. The message ‘*Your record has been made permanent*’ was displayed showing the correct working of the data entry form. ‘*Exit*’ button has been pressed to exit the form.

6.5 Testing Modify forms

I have tested the Modify form for books to check its validity. All the modify forms have the same action steps. The form initiated with the field for entering Accession number of the book whose record was to be modified. The screen has a button ‘*Show*’ and ‘*Select from list*’. I have selected the Accession number from the List of Value appeared by pressing the button ‘*Select from list*’. Then ‘*Show*’ button was pressed that gave me the record displayed in the form. I tried to change the Accession number of the book but an error message of ‘*You cannot change the record*’ has been displayed by the system. Thus showing the validity of the system. I have changed the ‘*Pages*’ of the book. Then ‘*Save*’ button was pressed for saving the changes. I pressed the ‘*Search*’ button for displaying another record for modification. After confirming modifications ‘*Exit*’ button was pressed to exit the form.

6.6 Testing Acquisition Forms

The Acquisition forms for books and serials have the same steps of action. Therefore the Acquisition form for books was tested for validations. I have entered all the empty fields. The publisher id and supplier id have been selected from the list of values provided by the buttons '*Select from list*'. The '*Modify*' button has been pressed to modify the publisher id in the '*Acquired Books*' block. I tried to alter the title name but error message was generated by the system. A message '*For placing order, click the title and press the button*' was generated by the system. I clicked a title and press the button of '*Place order*'. Another form '*Order book*' has been opened.

In the Book order form the order number has been generated as the sequence number. There was a tooltip '*If you are coming from acquisition, then click the title field*' with Title field. I clicked the title field, the title and author of the book previously selected in the acquisition form appeared. I entered the number of copies and supplier number. I pressed the '*Save*' button for saving the order placed. For received orders, the check box was checked. The fields for individual price, invoice number and payment were enabled for entry. The entries have been made and record was saved in the database thus showing the validity of form.

6.7 Testing Issue book form

The issue book form has been tested by entering the fields provided to the user. The Accession number has been inserted in the empty field. I tried to insert a number but an error message '*Book already issued*' appeared, showing that the book entry I was trying to make has already issued to someone else. By entering the right entry, member id was entered but an error message '*Member has left*' appeared, showing that the entry is incorrect. After entering the correct entry, the member type of the particular member was generated by the system. The issue date has been generated by the system as System date. On double clicking the '*Due date*' field I got the required due date specified for the

particular membership type. 'Save' button saved the record entered. On double clicking the member id, the list of the books already issued to that member appeared in the next block.

6.8 Testing Return Book form

The form for Return Book was tested through various steps. The form has been initiated with the option of entering the Accession number of the book either by selecting from the list or by user entry. I entered the number and pressed the 'Show' button. The record of the book issued to the Particular member appeared. The other block was displaying the other books already issued to the member along with the calculated fine. I entered the return date with the format given. On pressing the 'Enter' button, I got the calculated fine that has been due to the member. On entering the 'Paid' field I got the balance amount.

In case of damaged book, a block has been provided. I checked the check box of 'Damage'; the radio buttons for 'Recovery' were enabled. I selected the 'Rebind' option and entered the amount charged with the payment made. The 'Save' button got my changing saved to the database.

6.9 Testing Reports

The reports have been tested by running them from the main menu as well as from the forms where the 'Print' option has been provided. The reports of 'Available books' was opened from the 'Main Screen', the report appeared with an error log showing the successful opening of report.

APPENDIX - A

USER GUIDE

A. USER GUIDE

The user guide contains the layout of forms implemented by the tool Form Builder. The interface and functionality of the forms are described in the appendix for the user's convenience.

A.1 User Validation

The system starts with the user authentication by giving User Name and Password.

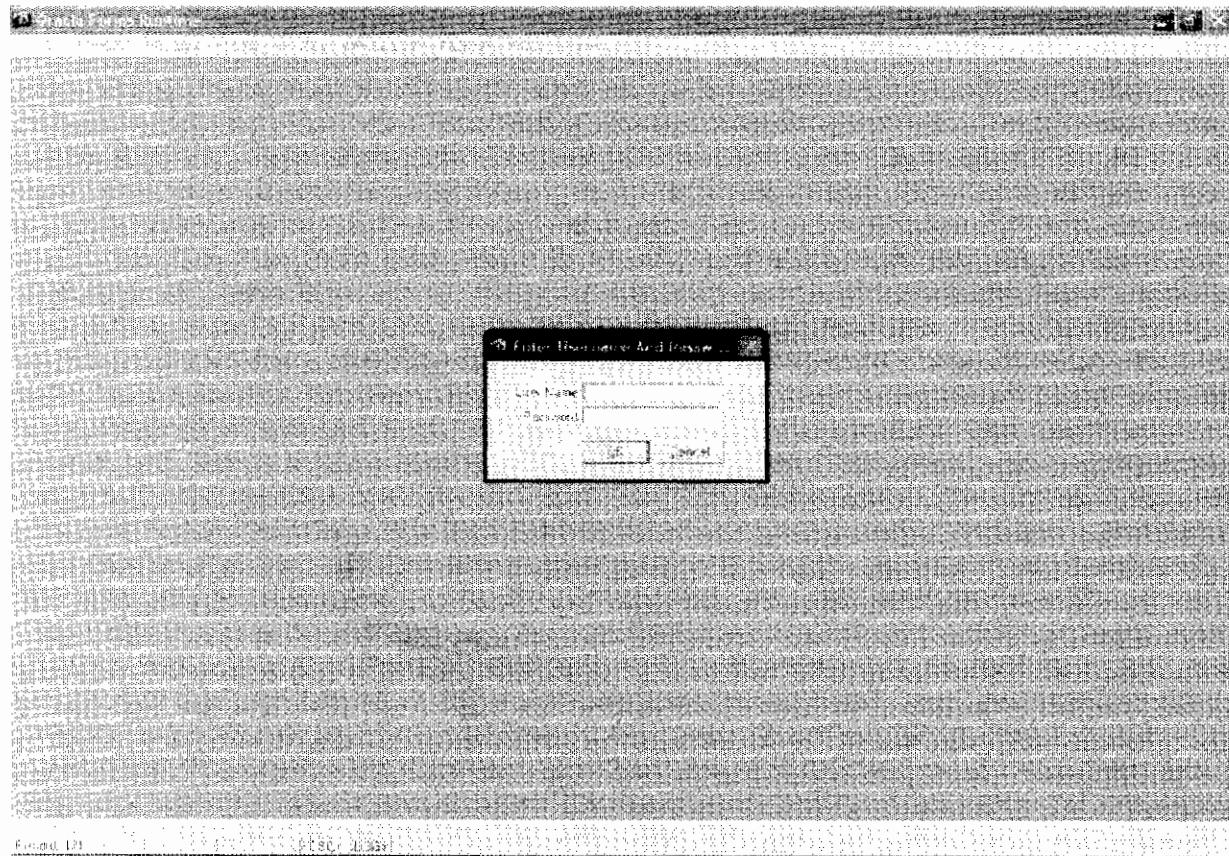


Figure A.1 User Validation form

A.2 Main Screen

This is the main screen providing a Menu for using other forms and reports. Every time the user exits from any form, control comes back to this screen.

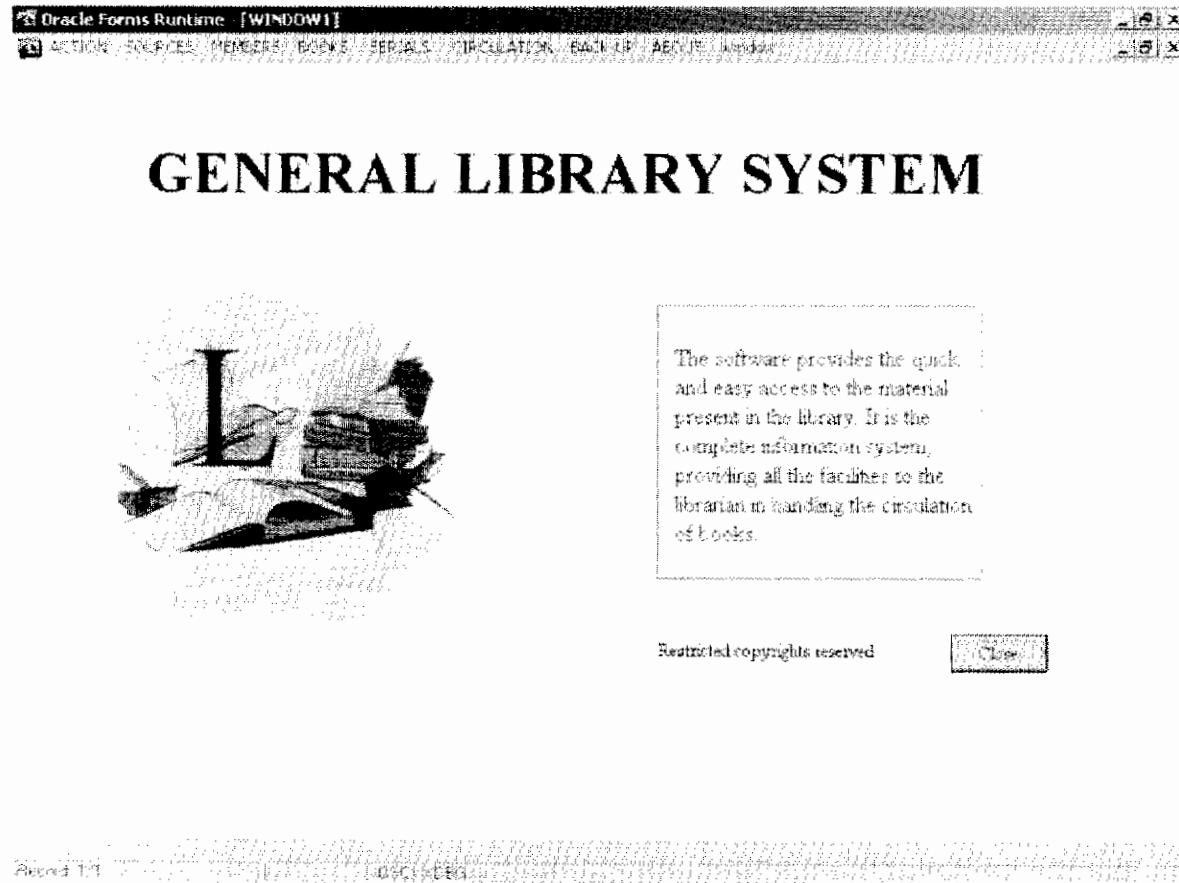


Figure A.2 Main screen of General Library System

A.3 Menu At the Main Screen

For every further process, the user drives the Menu. This menu is provided with all forms.

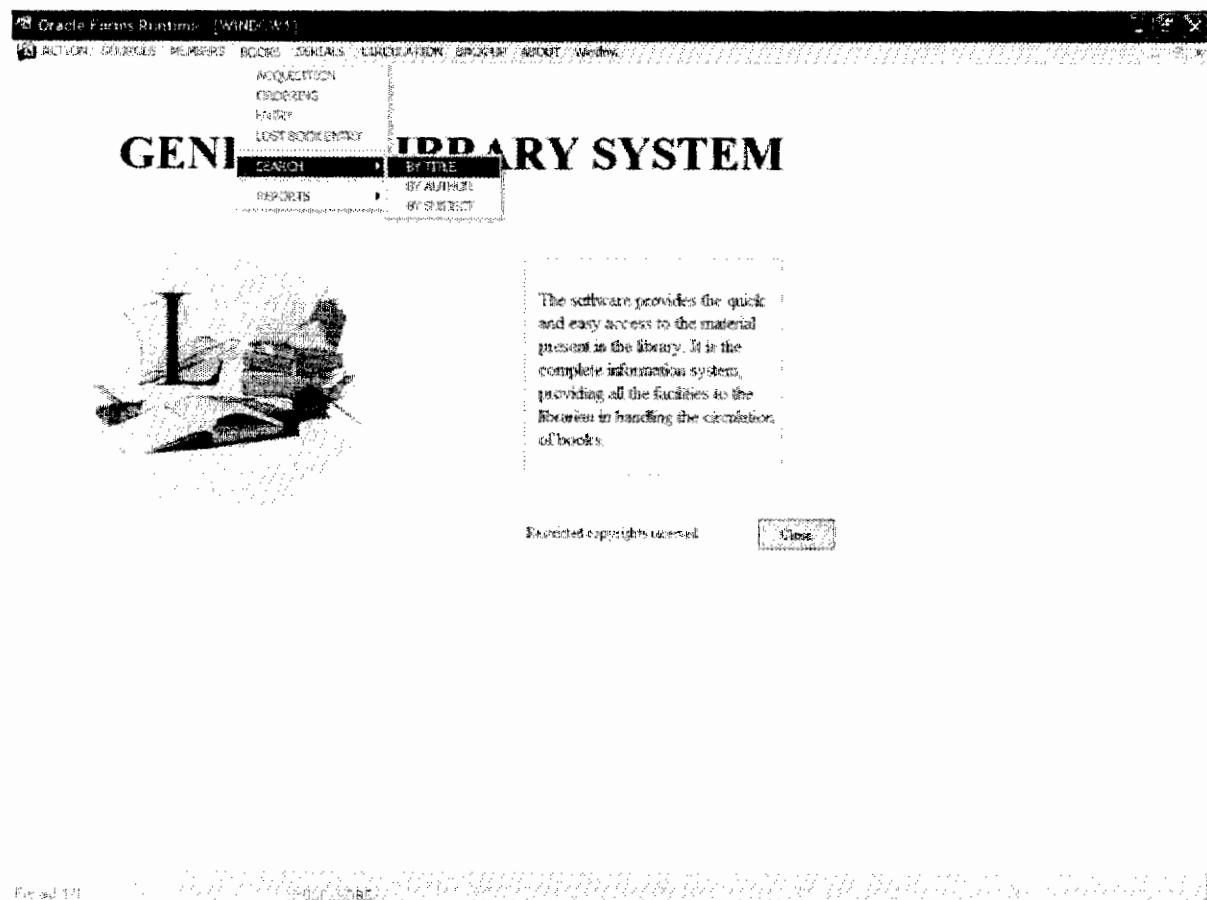


Figure A.3 Menu at main screen

A.4 Books Entry Form

This form is used for entering the detail information about the books. Accession number is generated automatically by the system. By clicking 'Show Purchased Book' button user will be provided the books purchased by the library. Clicking button 'Select from list' provides the list of values for Publishers and Suppliers.

Figure A.4 Books entry form

A.5 Modify Books Detail Form

This is a form used to modify the books details. The screen displayed contains the field for entering Accession number of the books whose record is to be modified. The accession number can also be selected from list. On pressing 'Show' button the record will be displayed.

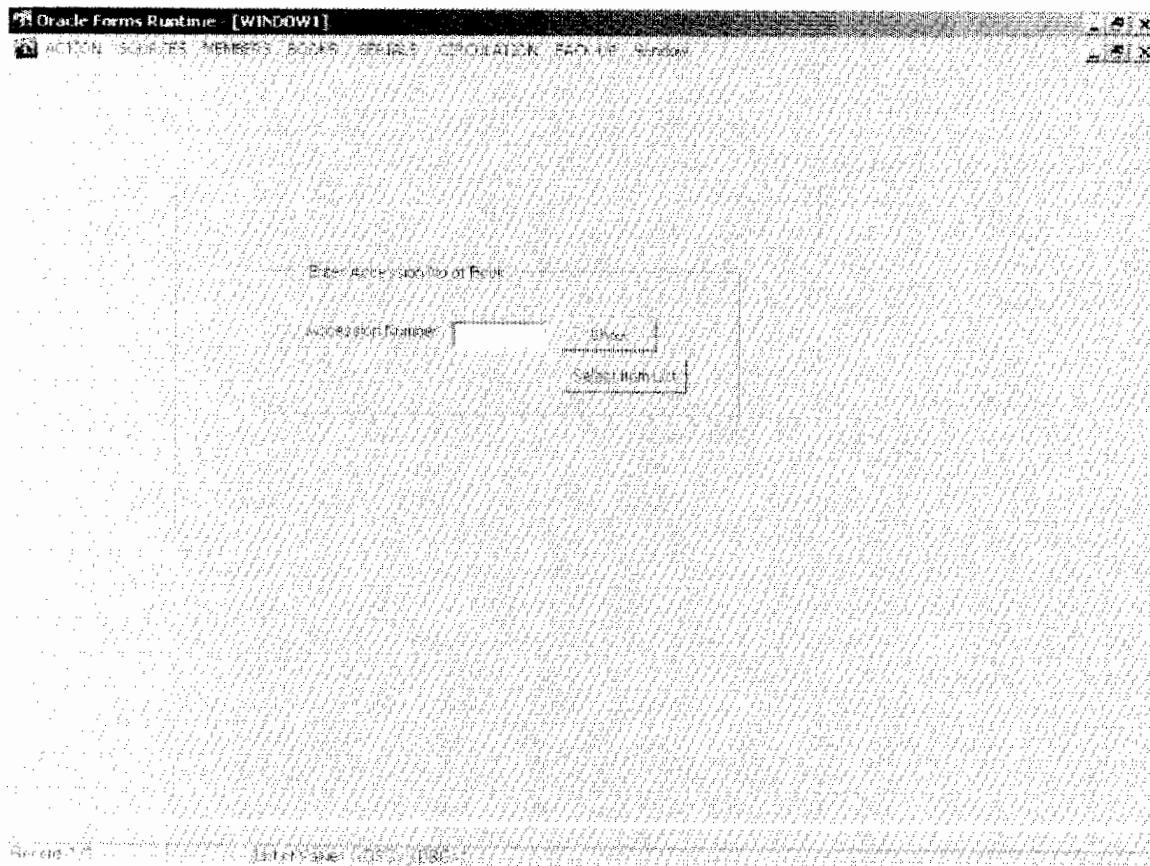


Figure A.5 Start screen of Books modify form

A.6 Modify Books Detail Form

(On clicking Select from List)

The accession number can also be selected from list. On pressing ‘Show’ button the record will be displayed.

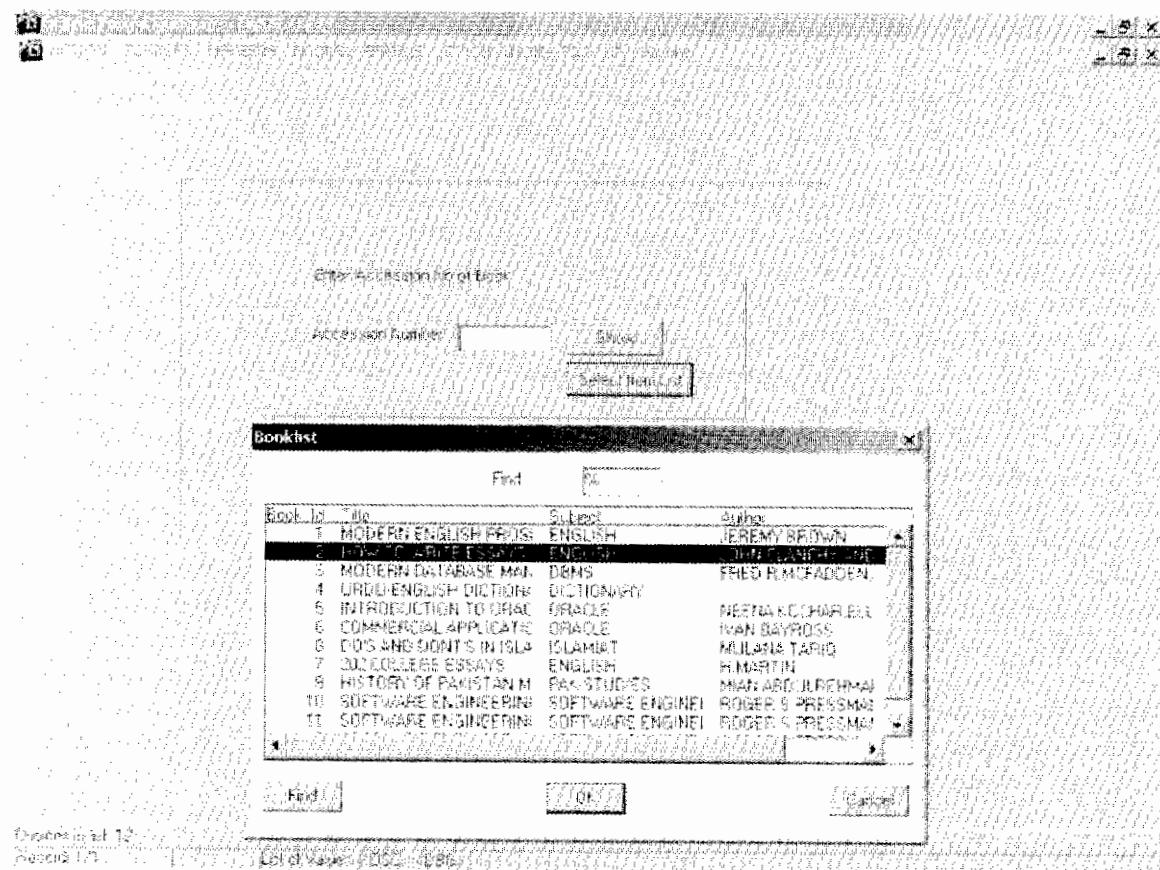


Figure A.6 List of value display on clicking ‘Select from list’

A.7 Modify Books Detail Form

The form gives the record of the particular book. The Accession Number of a book cannot be modified along with some other fields. After the modification clicks 'Save' button to save the modifications made to record.

Oracle Forms Runtime - [WINDOW1]

ACTION: CATEGORIES MEMBERS BOOKS SERIALS CIRCULATION EXIT-LS WORKS

Books View\Modify Form

Books

Accession ID: 1 Status:

Publisher ID: 1 Book Name: Publisher Name: FERDOSIKA PVT. LIMITED

Subject ID: 2 Series Name: Another Subject Name: MAYRA BOOKS

Subject: DICTIONARY

Author: ENGLISH-URDU DICTIONARY

Title: A COMPREHENSIVE DICTIONARY OF CURRENT

Page No: Page No: Editor:

Publication: 01/98 Year: 1998 Place: LAHORE PAKISTAN Volume No: 1

ISBN: 959005081 Pages: 631 Price: 198

Barcode No: 102

Reference: Location Shelf: FLOOR 10

Notes: DICTIONARY FOR REFERENCE

Shaped
 Torn
 Damage

Page: Page: End:

Figure A.7 Record display for modification

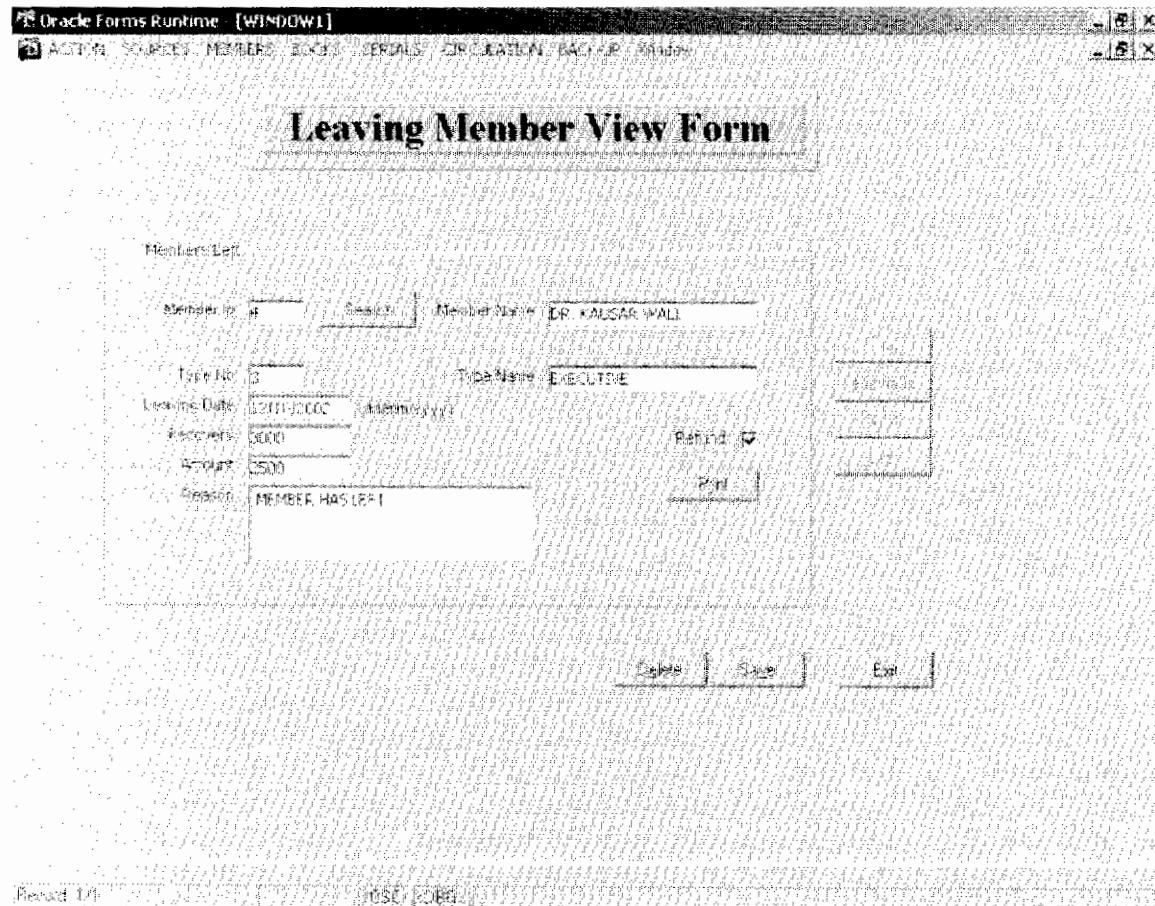
A.8 Leaving Member Entry Form

The form is used to enter the record of member leaving the library by canceling the membership. After entering the member id, by double clicking the field will give the details of books already issued to him that has not been returned.

Figure A.8 Leaving Member entry form

A.9 Leaving Member View/Modify Form

By entering the member id whose data is to be modified, the user can view the record. 'Print' button shows the printed report of members left.



Oracle Forms Runtime [WINDOW1]

Leaving Member View Form

Members Left

Member ID: 101 Member Name: DR. FAISAL WANI

Type ID: 13 Type Name: EXECUTIVE

Leaving Date: 12/31/2000 Amount: 5000

Reason: MEMBER HAS LEFT Period: 1

Print

Cancel Close Exit

Period: 1

Figure A.9 Form displaying record of member for modification

A.10 Lost/Damaged Books Entry Form

The form used for entry of books lost or damaged by the member or by any other source. If book is lost by any other source, then the field of member id will be disabled. In case of 'Status = Lost' the 'Recovery' will be 'Yes' or 'No', on the other hand in case of 'Status= Damage' the Recovery will be 'Replace, Rebind or Replace'. Books original price will be displayed by double clicking the field.

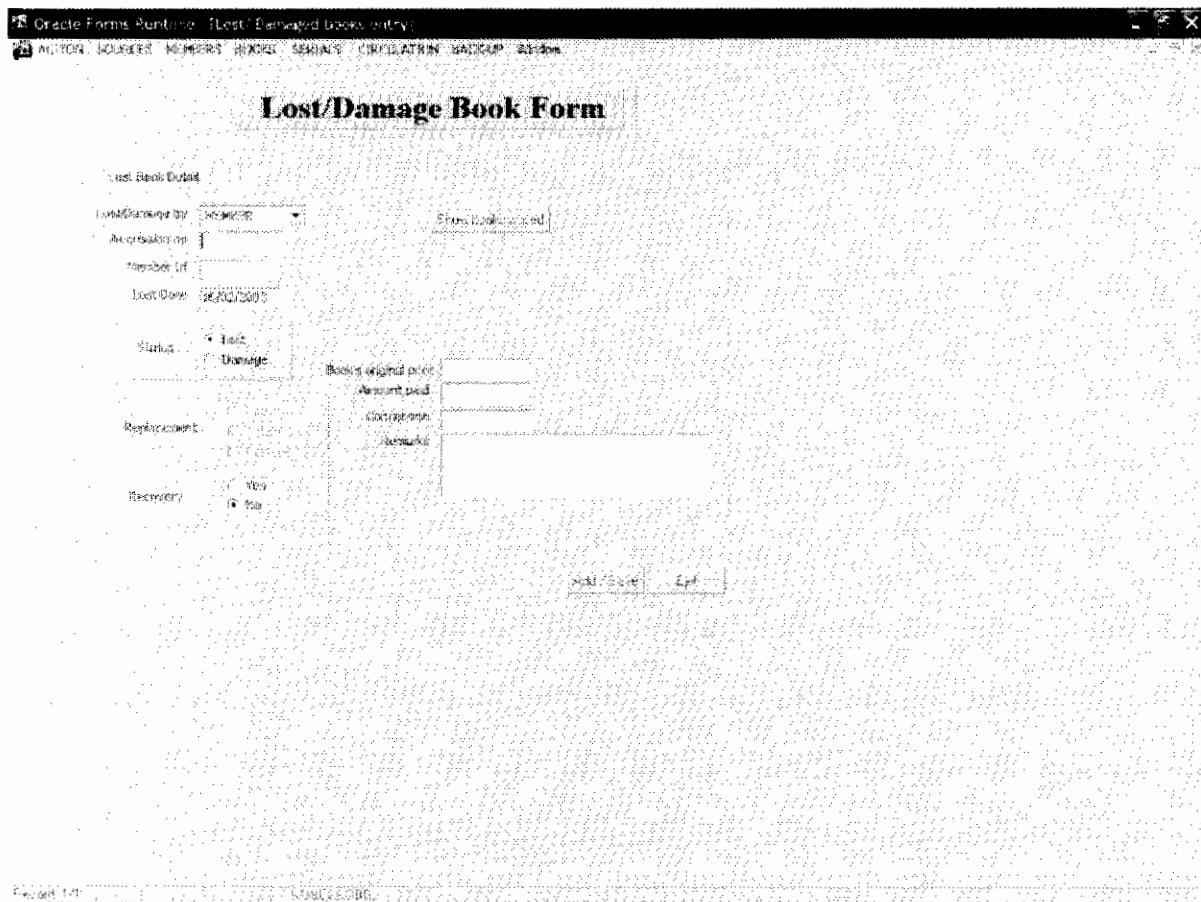


Figure A.10 Form for entering lost/damage books

A.11 Lost/Damaged Books Modify Form

The lost/damaged books can be modified using the same technique as described for the other modify forms.

The screenshot shows a window titled 'Lost/Damage Books' from the Oracle Forms Runtime. The window has a menu bar with 'ACTION', 'SOURCES', 'MEMBERS', 'BOOKS', 'SQL/PLSQL', 'COLLECTIONS', 'BOOKCAT', 'PROGS', and 'Help'. The main area is titled 'Lost/Damage Books' and contains the following data:

Lost/Damaged By	MEMBER	Book Title	Author Name
Accession Number	SEARCH	HOW TO WRITE ESSAYS	AMITRAJAHNOO
Member ID			
Issue Date	23/10/2002		
Book Status	LOST	Books Availability	YES
		Amount Due	RS
		Penalty	0
Replaced Book	NO	Previous Status	LOST BY MEMBER
Recovery	YES		

At the bottom of the form, there are buttons for 'Delete', 'Save', and 'Exit'.

Figure A.11 Form displaying record of lost book for modification

A.12 Member Type Policy Entry Form

The form is used to enter the details for membership category and policy applied for the specific category. Type number is generated automatically by the system. The members are issued books keeping in view the policy of their member type.

The screenshot shows a window titled 'Member Type Policy Form' from the 'Oracle Forms Runtime (WINP001)' application. The window has a menu bar with 'FILE', 'EDIT', 'VIEW', 'ACTION', 'SOURCES', 'MEMBERS', 'BOOKS', 'CERTAINS', 'EXTRACTION', 'GRANT-ER', and 'IMAGES'. The main area is titled 'Member Type' and contains the following fields:

Type No.	Type Name
Issue Per Day	Days of Utilization
Max Books Allowed	Stop Date
Validity	Days after validity or stop after issue date

Below the table, there is a note: 'Valid only if Stop Date is not specified'. At the bottom of the form, there is a 'Save' button and a 'Cancel' button.

Figure A.12 Form for entering types of membership

A.13 Member Type Policy View/Modify Form

The form is used to view and modify the member type policy details. The user can also view the list of members belong to any specific member type.

Oracle Forms Runtime - [WINDOW]

Auton Source: MEMBERS FORMS APPLICATION FORM BY WINDOW

Member Type Policy Form

Member Type

Type ID	Specify	Member Type
1	STUDENT	
Fine Per Day	Days of late	
Min. Books Available	Max. Fine	
Books Due	Days for late return	
	of the after due date	

Showing Members

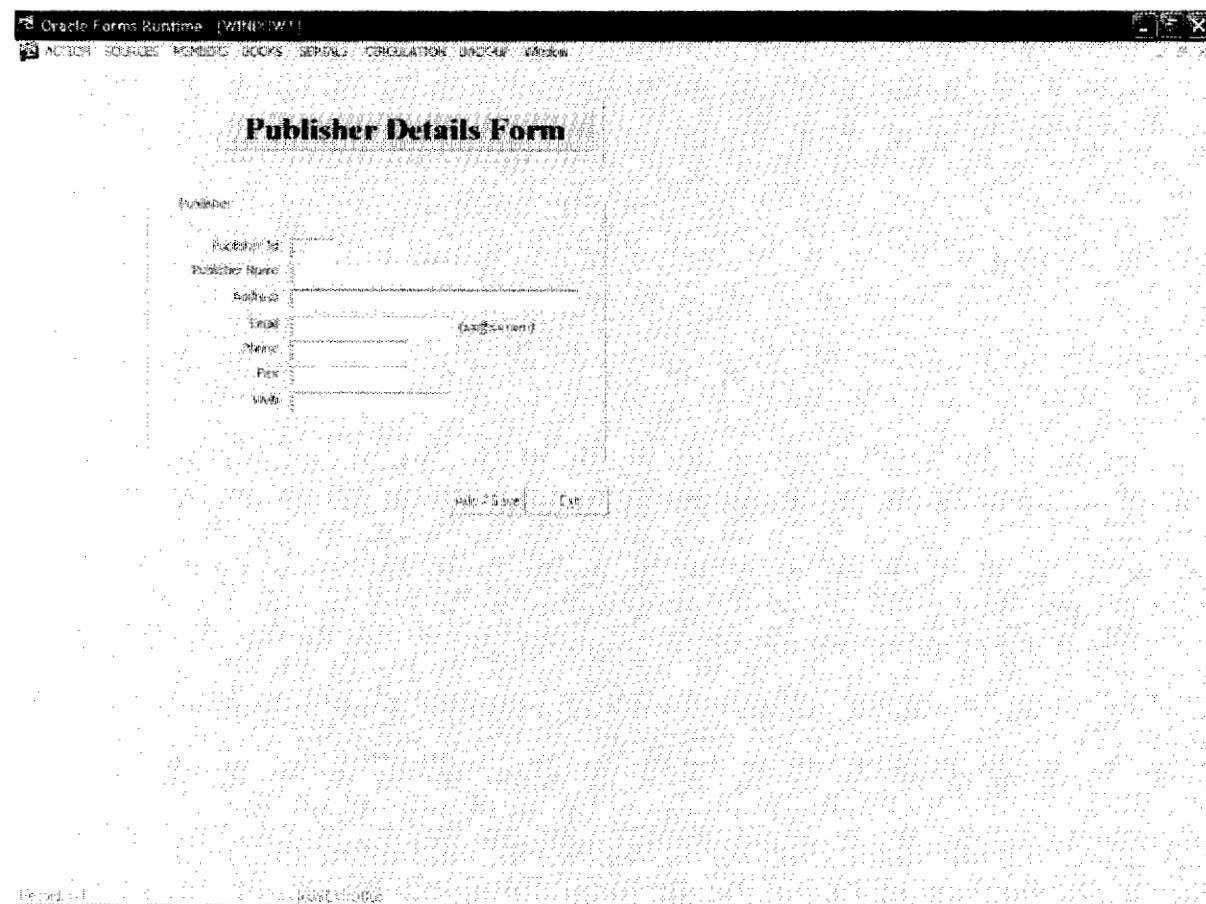
Member ID	Name	Ref Name	Address	Phone	DOB	Gender	Books	Days
1	AMNA MAHMUD	RISHAD MAHMUD	39-A SATELLITE TOWN GUJ 031 424433	18-DEC-2004	F	1000	0	
2	HASAN MAHMUD	RISHAD MAHMUD	39-A SATELLITE TOWN GUJ 031 292244	12-DEC-2004	M	1000	100	
3	KARMAI SALMA	MUHAMMAD SALMAN	H-51 G-10-3 ISLAMABAD 051-4422324	12-SEP-2003	M	1000	0	
4	PRIMA	SPK	00F	03322322	13-DEC-2002	F	1000	0

Page 1/1

Figure A.13 Modify membership type form

A.14 Publisher Details Entry Form

The publisher details are entered here for being available in other forms as reference. Publisher id is generated by the system automatically.



The screenshot shows a window titled "Publisher Details Form" from the Oracle Forms Runtime (WIN32W). The window has a menu bar with the following items: ACROSS, SOURCES, HOMELOC, BOOKS, SERIALS, CIRCULATION, LIBRARY, and ABOUT. The main area of the window is a form titled "Publisher". The form contains the following fields:

Publisher	
Publisher Id	
Publisher Name	
Address	
Email	lalit@sanchari
Phone	
Fax	
Web	

Below the form, there is a status bar with the text "Entered 11/11/2008 10:55:00 AM" and a "Cancel" button.

Figure A.14 Form for entering Publishers

A.15 Publisher Details Modify Form

The publisher details can be modified using the same process as describe earlier.

Oracle Forms Runtime - [WINDOW1]

ACTION SOURCES MEMBERS BOOKS SERIALS CIRCULATION BACK-UP HISTORY

Publisher Details Form

Publisher

Publisher ID:

Publisher Name: SHRI GANESH PUBLISHERS AND CO.

Address: 111 BURJ MEHRAULI CUTTA

Email: sgp@sgp.com

Phone: 01222631572

Fax: 01222633651

URL:

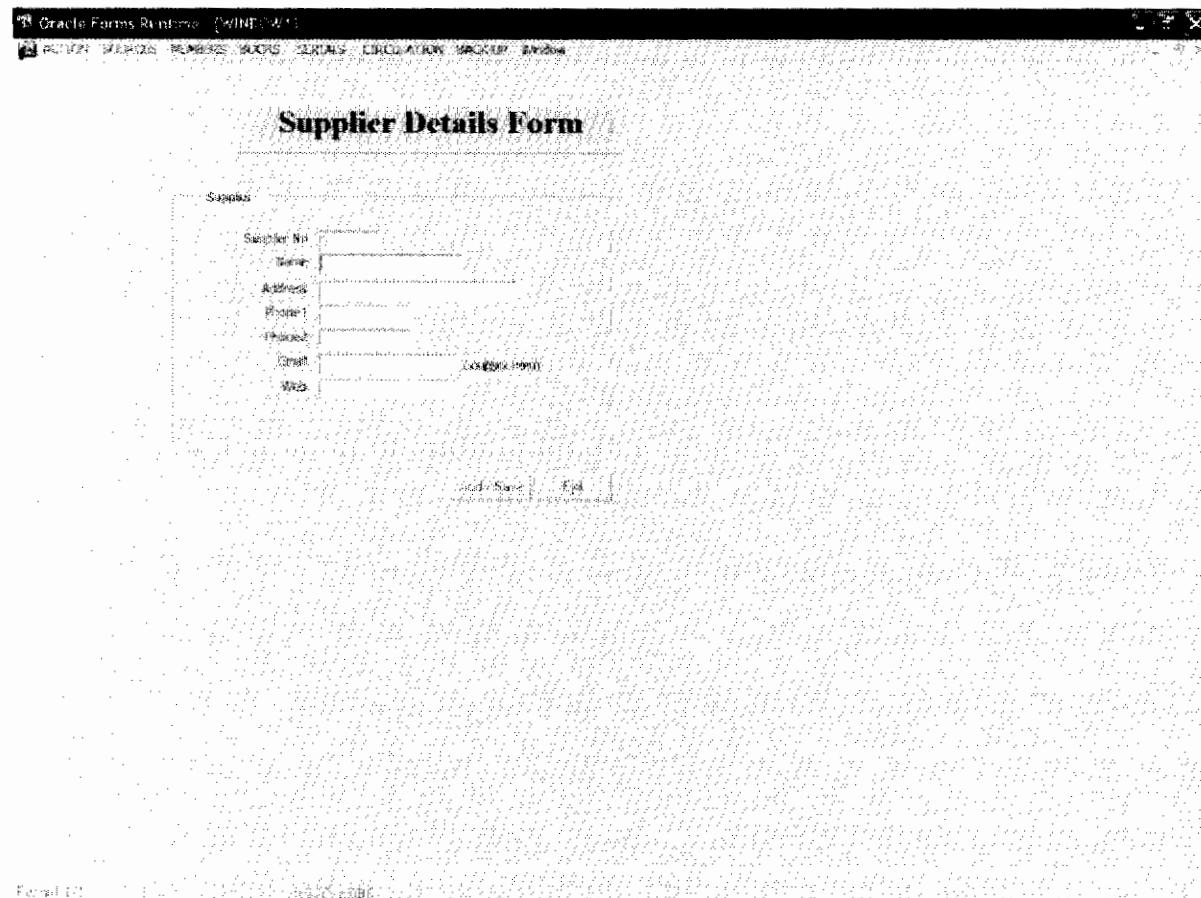
Save Cancel Exit

Record 1/1

Figure A.15 Publisher's Detail modification form

A.16 Supplier Details Entry Form

This form is used to keep record of the suppliers and vendors who provide books and serials to the library either for selection or in case of orders. The record is used to find out the address of any specific supplier for placing orders.



A screenshot of an Oracle Forms Runtime application window titled "Supplier Details Form". The window has a menu bar with "FILE", "EDIT", "HELP", and "SUPPLIER". The main area contains a form with the following fields:

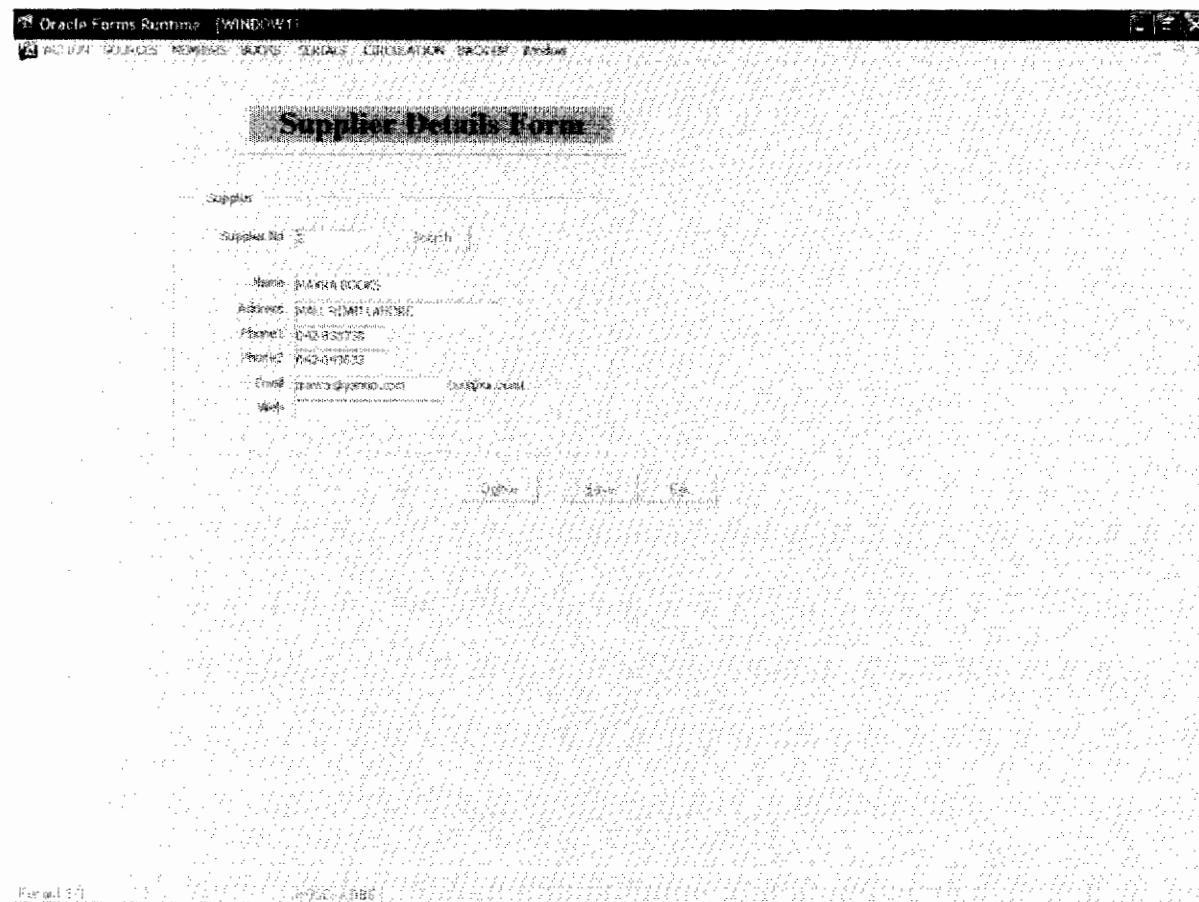
Supplier	
Supplier No	1234567890
Name	ABC Books
Address	123 Main Street, Anytown, USA
Phone	(555) 123-4567
Fax	(555) 123-4568
Credit	0000000000000000
Web	http://www.abcbooks.com

Below the form are "Add", "Save", and "Exit" buttons. The status bar at the bottom shows "Form1 (1234567890)" and "Supplier Details Form".

Figure A.16 Supplier's detail entry form

A.17 Suppliers Detail Modify Form

The form is initiated by entering the supplier name whose data is to be modified. The record after modification is saved.



The screenshot shows a window titled "Supplier Details Form" from the Oracle Forms Runtime. The window has a menu bar with "FILE", "EDIT", "WINDOWS", "HELP", and "ABOUT". The main area is titled "Suppliers" and contains the following data for a supplier named "MANNA BOOKS":

Supplier No	Search
1	
Name	MANNA BOOKS
Address	PO BOX 555 LAUREL
Phone	042-857738
Phone2	042-419833
Email	pranav@yahoo.com
Web	

At the bottom of the form, there are buttons for "Cancel", "Save", and "Close".

Figure A.17 Supplier's detail modify form

A.18 Books Acquisition Form

The form is used to keep the record of books received by the library for selection. If the authorities reject the books, the checkbox is checked in the Books Acquired block. Record can be deleted from the Books Acquired block. After selection, order can also be placed by clicking the button 'Place Order', thus invoking another form for order placing.

Oracle Forms Runtime - Books Acquisition

BOOKS SOURCES MEMBERS BOOKS SERIES CIRCULATION BACCODE WORKS

Books Acquisition Form

Books Acquisitions

File	Source	Author	Publication Date
Author	Publication ID	Author	Source ID
Publisher	Source No.	Author	Source ID
ISBN	Source ID	Author	Source ID
Acquisition type	Book Order		
	Ex		

Books Acquired

Book ID	Title	Author	Publisher ID	Year	Acquisition type	Quantity	Supplier ID
114-001-0022	THE STAN FORD TALES	MARK TAUBER	4	1999	PURCHASE	1	SUPPL002
114-001-0023	THE LEONARDO	JOHN TAUBER	4	1999	PURCHASE	1	SUPPL003
114-001-0024	CHILD OF THE REAMS	ROBERT RODIN	2	1998-1999	PURCHASE	1	SUPPL004
114-001-0025	MECHANICAL PHYSICS	DAVID MARTIN	3	1998-1999	PURCHASE	1	SUPPL005
114-001-0026	MORE ARNE MORE ERNE	PIERRE LEGRAND	4	1998-1999	PURCHASE	1	SUPPL006
114-001-0027	INSTITUTIONAL ANALYSIS AND DEVELOPMENT	EDUARDO MUNOZ	2	1998-1999	PURCHASE	1	SUPPL007

Finishing order click the the amplified button
Record 111

Figure A.18 Form used for entering/viewing acquired books

A.19 Order Books Form

This is the form appeared by clicking 'Place Order' in Acquisition Books form. Single clicking the field of Title gives the title and author automatically whose order is being placed. When the library receives order, the check box is checked in Books Received block and invoice number along with the payment made is entered.

Book Orders Form

Order Info

Order Info

Books Received

Order No.	Title	Author	Supplier No.	No. Of Copies	Order Received Date	Invoice No.	Amount
1	DR. DANTI	RABINDRANATH TAGORE	1	2	2020-01-01	22	100
2	LOST IN THE FOREST	ROBERT BROWN	2	2	2020-01-01	23	100
3	THE JUNGLE	JOHN GALT	3	2	2020-01-01	24	100
4	ONE DAY	DAVID KORULIDZE	4	2	2020-01-01	25	100
5	BYETEM ASYL, GOS AND DE SELAS MIRANDA	5	2	2	2020-01-01	26	100

Figure A.19 Form for entering/viewing orders placed

A.20 Search By Author Form

This form is used to search any particular book by entering the name of Author in the field. Pressing the button 'Search' gives the list of books of the same author. Pressing the button 'Print' can print this list.

The screenshot shows a window titled 'Search By Author Form'. At the top, there is a menu bar with 'ACTION', 'SOURCES', 'MEMBERS', 'BOOKS', 'SERVICES', 'CIRCULATION', 'BACKUP', and 'Help'. Below the menu, there is a search bar labeled 'Search by Author' with the value 'Author: PRESSMAN'. To the right of the search bar is a 'Search' button. The main area is titled 'Books Available' and contains a table with the following data:

Book No.	Subject	Title	Author	Edition	Status	Location	REF
10	SOFTWARE ENGINEER	SOFTWARE ENGINEERING: A PRACTITIONER'S APPROACH	ROGER S. PRESSMAN	4	P	CARENS	A
11	SOFTWARE ENGINEER	SOFTWARE ENGINEERING: A PRACTITIONER'S APPROACH	ROGER S. PRESSMAN	4	P	CARENS	A
12	SOFTWARE ENGINEER	SOFTWARE ENGINEERING: A PRACTITIONER'S APPROACH	ROGER S. PRESSMAN	4	P	CARENS	A

Figure A.20 For searching book by Author

A.21 Search By Subject Form

This form is used to search any particular book by entering the 'Subject' in the field. Pressing the button 'Search' gives the list of books of the same subject. Pressing the button 'Print' can print this list.

Figure A.21 Form for searching book by Subject

A.22 Search By Title Form

This form is used to search any particular book by entering ‘Title Keyword’ in the field. Pressing the button ‘Search’ gives the list of books having the same keyword. Pressing the button ‘Print’ can print this list.

The screenshot shows a window titled 'Search By Title Form' within 'Oracle Forms Runtime - [WINDOW]'. The window has a toolbar with buttons for Action, Source, Member, Print, Serial, Calculation, Failure, and Help. The main area has a label 'Search by Title' and a text input field containing 'MCLENN'. A 'Search' button is to the right. Below is a table titled 'Books Available' with columns: Acc No, Subject, Title, Author, Edition, Status, Location, and Ref. One row is selected, showing Acc No 1, Subject ENGLISH, Title MODERN ENGLISH PROSE, Author JEREMY BROWN, Edition 3, Status P, Location CABINET SHELF 1, and Ref. The table has 8 columns. At the bottom are buttons for Print and Exit.

Acc No	Subject	Title	Author	Edition	Status	Location	Ref
1	ENGLISH	MODERN ENGLISH PROSE	JEREMY BROWN	3	P	CABINET SHELF 1	
2	ITEMS	MODERN DATABASE HANDBOOK	FRED R. McFADDEN, JR.	1	P	CABINET SHELF 1	

Figure A.22 Form for searching book by Title

A.23 Issue Book Form

This form is used for Circulation Desk Processing. For issuing the book accession number of book is entered then the member id of the member is entered whose type number will be given automatically by the system. If the books accession number entered is already issued then an error message will be generated ' Book already issued'. Same is the case with member id whose membership has been cancelled or expired. Double clicking the Due date field will give the date automatically according to the member type policy. After entering member id, on double clicking the field, the user can get the list of books already issued to the member with fine calculation.

The screenshot shows the Oracle Forms Runtime interface on a Windows system. The title bar reads 'Oracle Forms Runtime (WINDOWS)'. The menu bar includes 'FILE', 'EDIT', 'VIEW', 'ACT/NIN', 'SOURCES', 'MEMBERS', 'BOOKS', 'SERIALS', 'CIRCULATION', 'BACKUP', and 'Window'. The main title of the form is 'Issue Books Form'. The form contains several input fields and a grid. The input fields are: 'Accession No.' (with a dropdown arrow), 'Member Id' (with a dropdown arrow), 'Member Type' (with a dropdown arrow), 'Issue Date' (with a date picker icon), and 'Due Date' (with a date picker icon). Below these fields is a grid titled 'Books Already Issued' with columns: 'Book Id', 'Book Name', 'Issue Date', 'Due Date', and 'Fine'. The grid has several rows of data. At the bottom of the form is a toolbar with icons for 'New', 'Edit', 'Delete', 'Save', 'Cancel', and 'Print'.

Figure A.23 Issue Book form

A.24 Return Book Form

The form is used to enter the return date of issued book when reached the library. On entering the return date if the book is overdue then fine will be calculated by the system. On payment balance will be there. On returning if the book is found damaged then its fine amount can also be entered on desk.

Figure A.24 Return Book form

A.25 Serials Entry form

This form is used to enter the new serial as well as the copies of already entered serials. The option for entering new serial or a copy is provided to the user at first screen of serial form. For adding a copy the user has to enter the title of serial or serial id assigned to the serial.

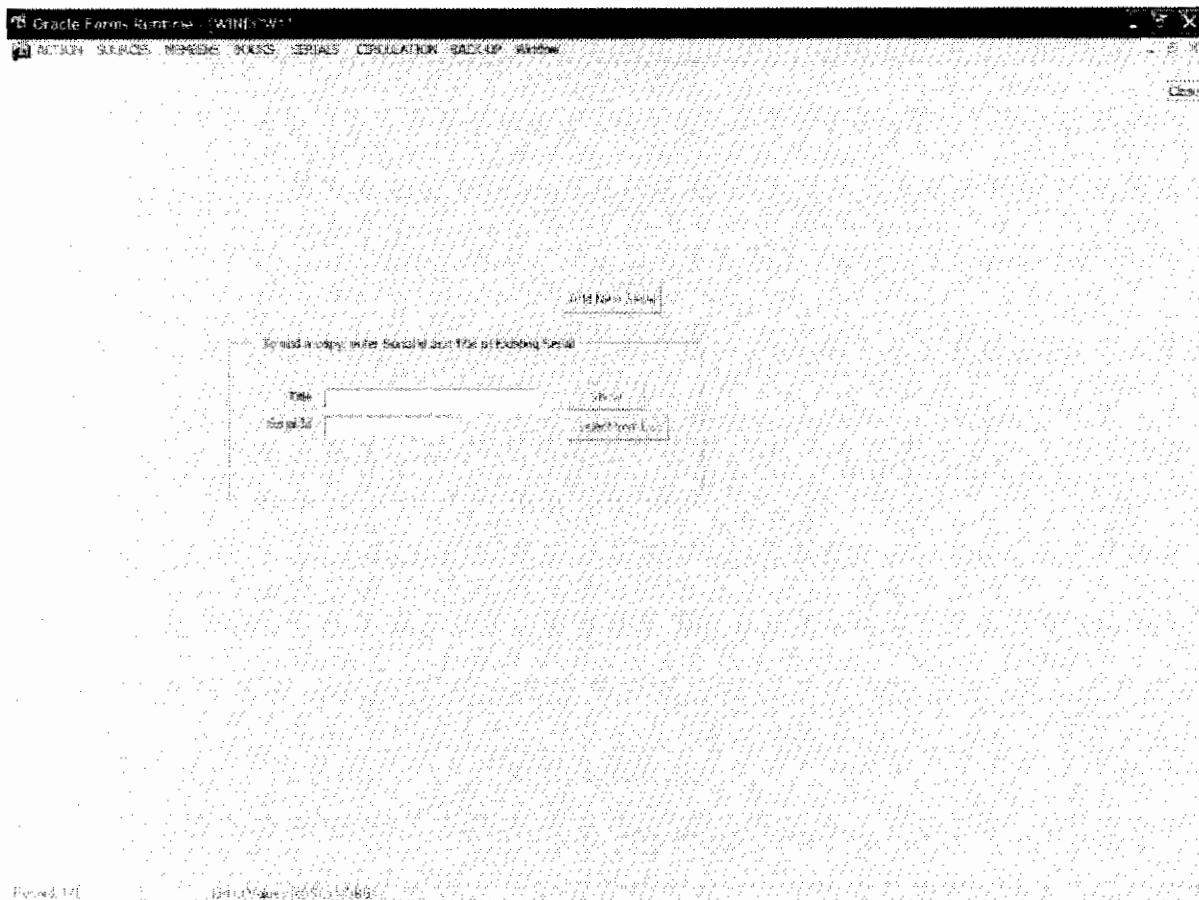


Figure A.25 Screen for choosing the option for serials

A.26 Serials Entry form

When the user clicks the button 'Add New Serial' from the starting screen, the serial is has been generated by the system and empty fields are there to make entry.

Figure A.26 Form opened for entering new serial

A.27 Serials Entry Form

(On entering title / selecting from List to add a copy)

When the user enters or selects the title of existing serial, then by pressing show button form is opened for adding the copy of the serial. Serial number is generated by the system as sequence number. 'Add a copy' button is also provided to add further copies.

The screenshot shows the 'Serials Entry Form' window. The title bar reads 'Serials Entry Form'. The form is divided into several sections:

- Serial Entry:** Contains fields for Serial ID (1), Title (Entertainment Weekly), Publisher ID (1), Language (ENGLISH), and Volume Number (1).
- Periodicals:** Contains fields for Entry Date (01-12-2012), Subject (ENTERTAINMENT), Title (Entertainment Weekly), and Title (Entertainment Weekly).
- Acquisition:** Contains fields for ISSN (1234-5678), Issue Date (20-12-2012), Frequency (WEEKLY), and Delivery Type (MAIL).
- Purchase:** Contains fields for Purchase Date (20-12-2012), Purchase Type (PURCHASE), and Purchase Price (100).
- Notes:** Contains a note about the serial being a monthly publication.

At the bottom, there are buttons for 'Add New Serial' and 'Add a copy'. The status bar at the bottom left shows 'Record 4/4'.

Figure A.27 Form for adding the copies of serial

A.28 Serials Acquisition Form

The form is used to keep the record of serials received by the library for selection. If the authorities reject the serial, the checkbox is checked in the Acquired Serials block. Record can be deleted from the Acquired Serials block. After selection, order can also be placed by clicking the button 'Place Order', thus invoking another form for order placing.

Period	Title	Subject	Item No.	Author	Publisher ID	Periodicals Right	Acquisition Type	Source	Supplier Selected
NO-DEC-2003	JADE MAGAZINE	WITH LATEST TRENDS	000-4731-17	A	MONTHLY MAGAZINE	PURCHASE	PURCHASE	SUPPLIER	
NO-DEC-2003	CRICKETER		030-3999-33	ALAM MAHABIR	M	MONTHLY MAGAZINE	ACQUISITION SUPPLIER		
03-DEC-2003	JOHNAH R. SAWAH		PAAS-4442-228		P	MONTHLY	PURCHASE	PURCHASE	

Figure A.28 Serials Acquisition Form

A.29 Order Serials Form

This is the form appeared by clicking 'Place Order' in Serials Acquisition form. Single clicking the field of Title gives the title and supplier automatically whose order is being placed. When the library receives order, the check box is checked in Orders Received block and invoice number along with the payment made is entered.

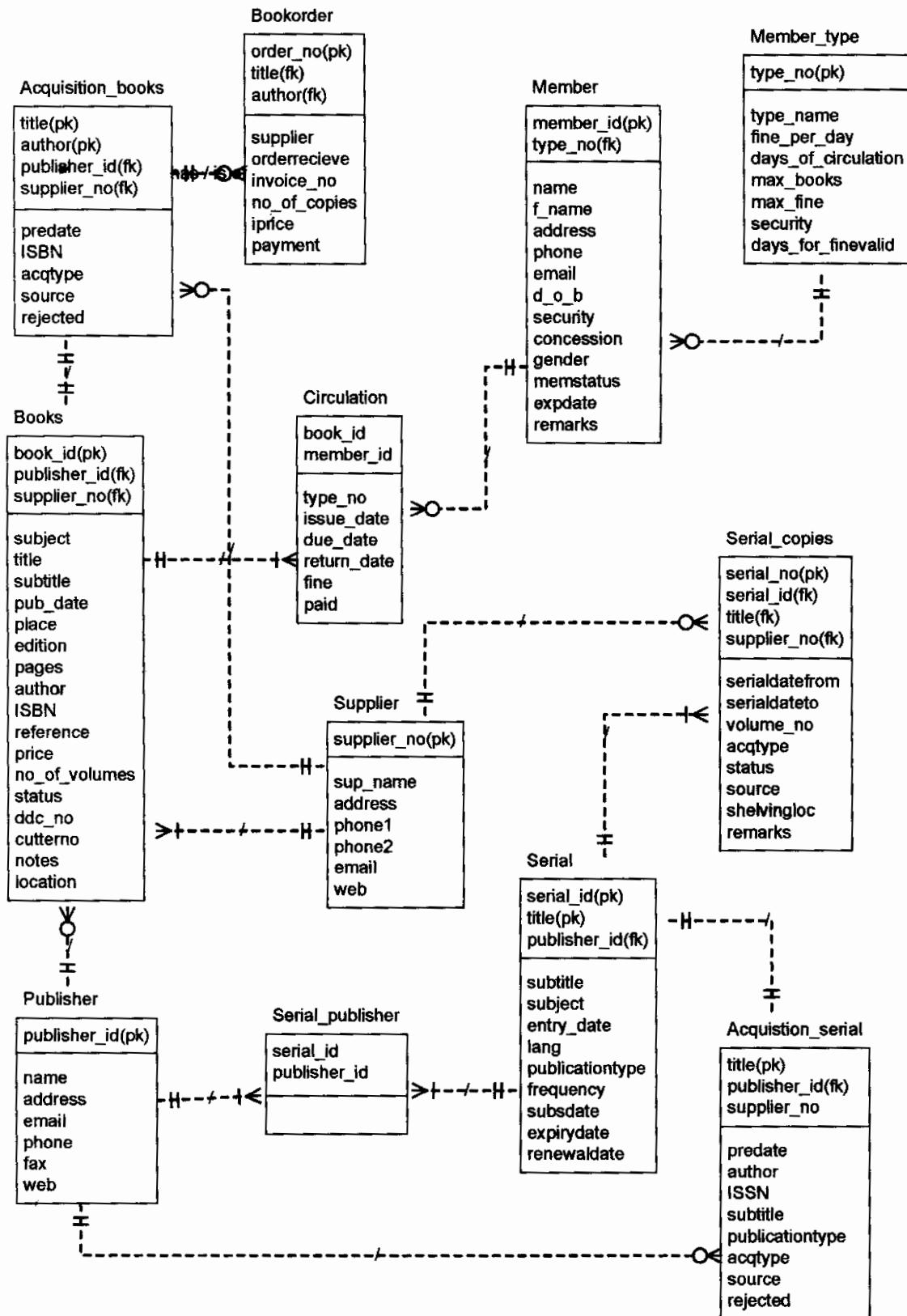
Order No.	Title	Supplier	Publisher	Ed.	Copies	Received	Turnaround	Invoice No.	Payment
1	ANGELA E. MATTIA	1	1	3		10-02-2003	132	500	
2	ANNE PANGATIBA	2	1	5		10-02-2003	132	500	

Figure A.29 Serials Ordering Form

APPENDIX - B

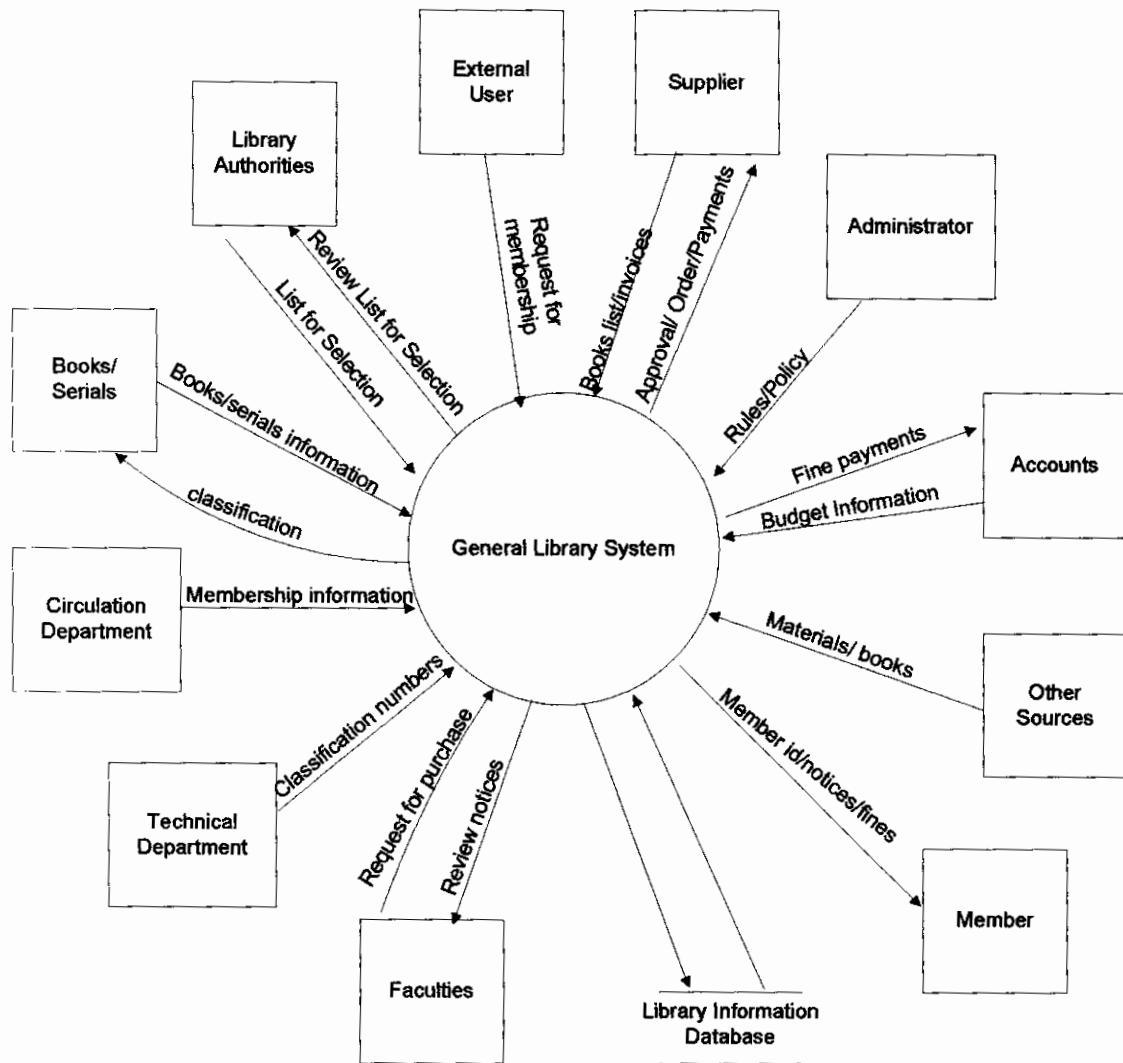
ENTITY RELATIONSHIP DIAGRAM AND DATA FLOW DIAGRAM

E-R Diagram of General Library System

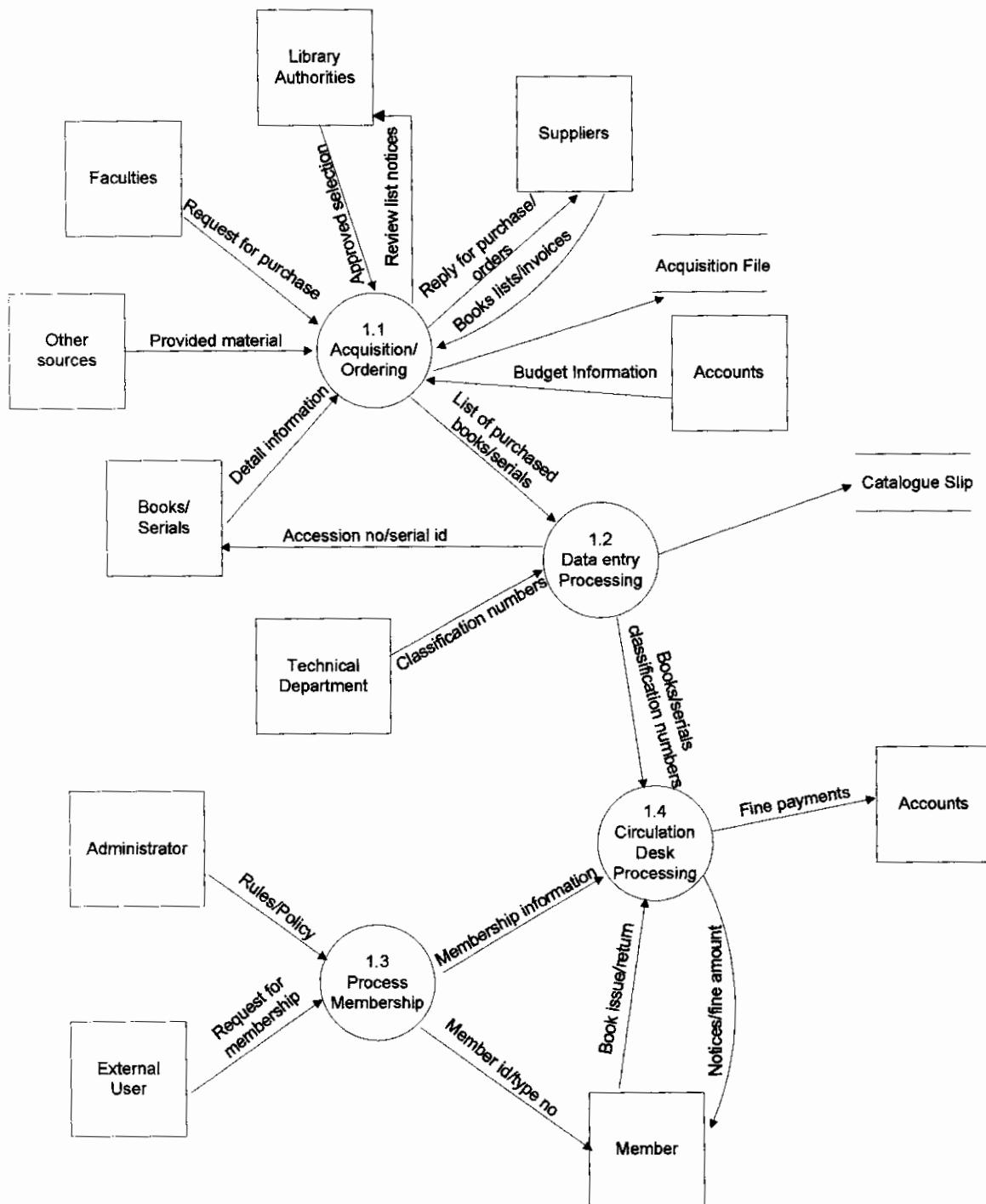


Data Flow Diagram of General Library System

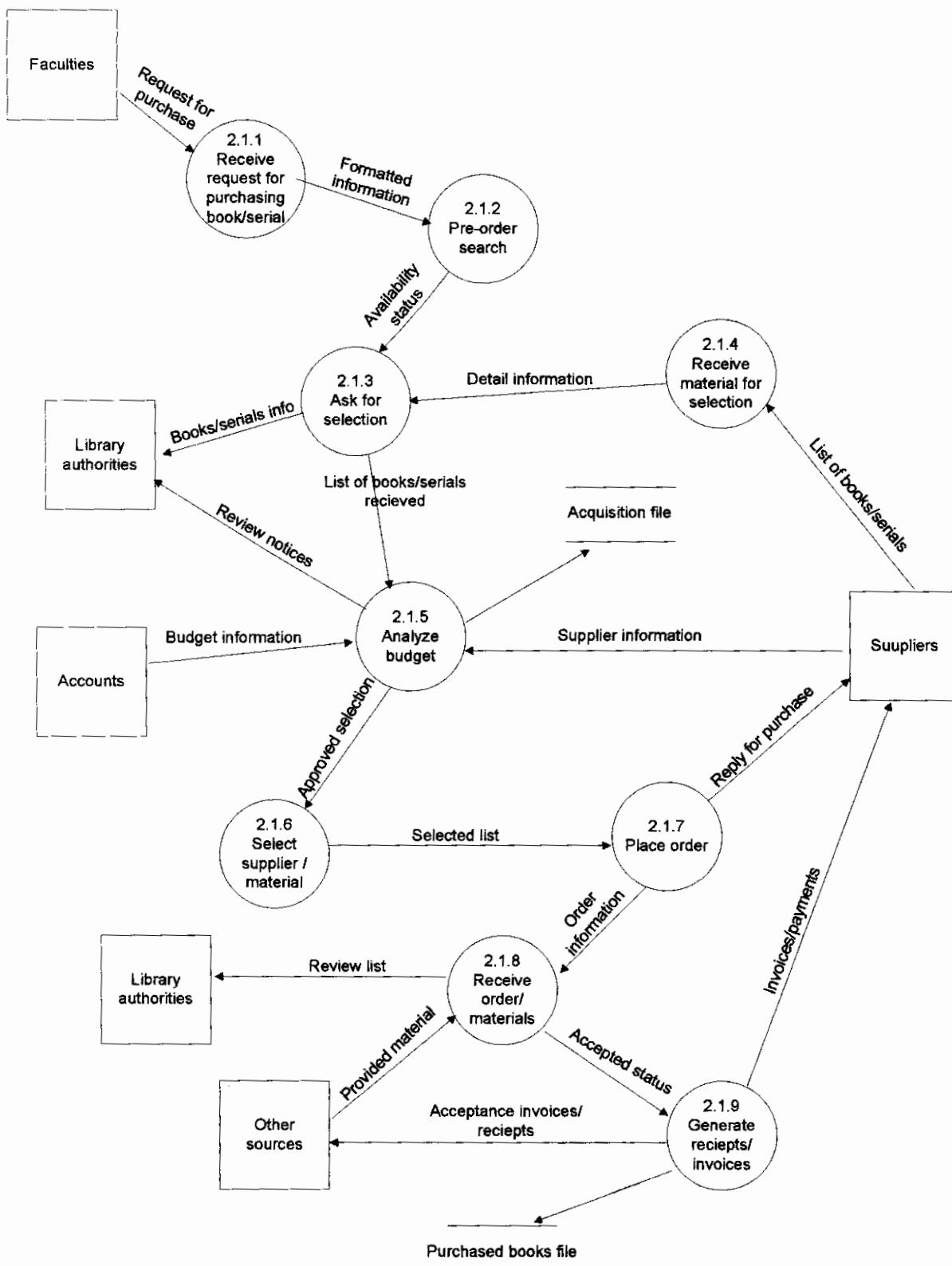
Context Level DFD



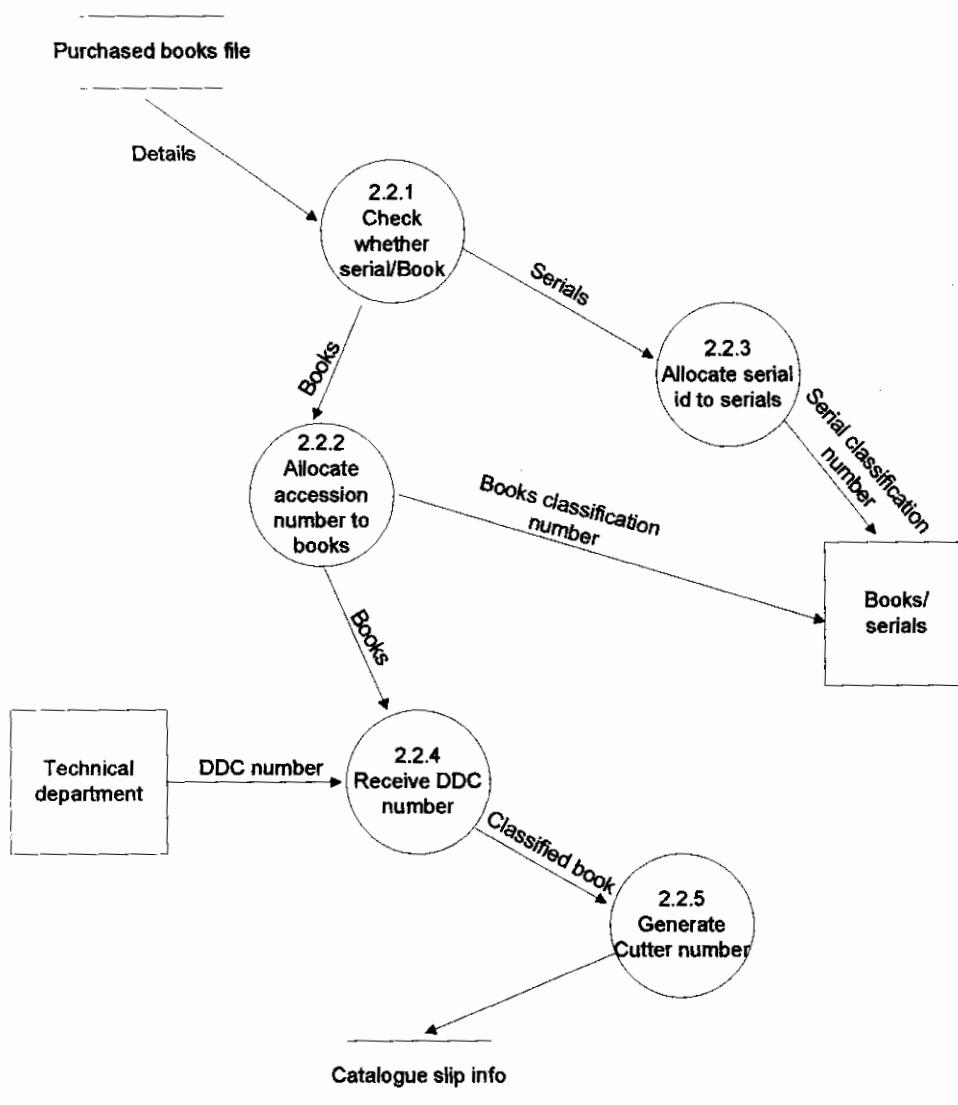
Level 1 DFD of General Library System



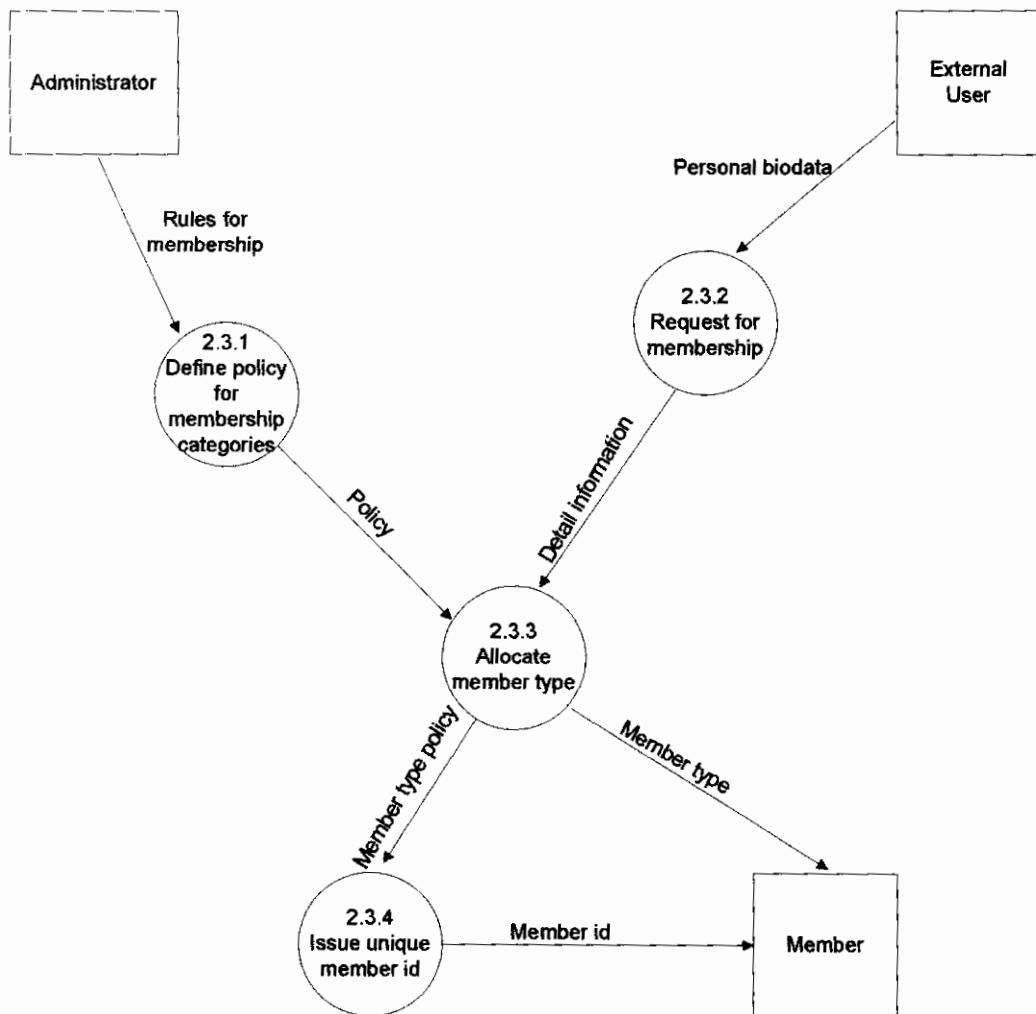
Level 2 DFD of Acquisition/Ordering Process



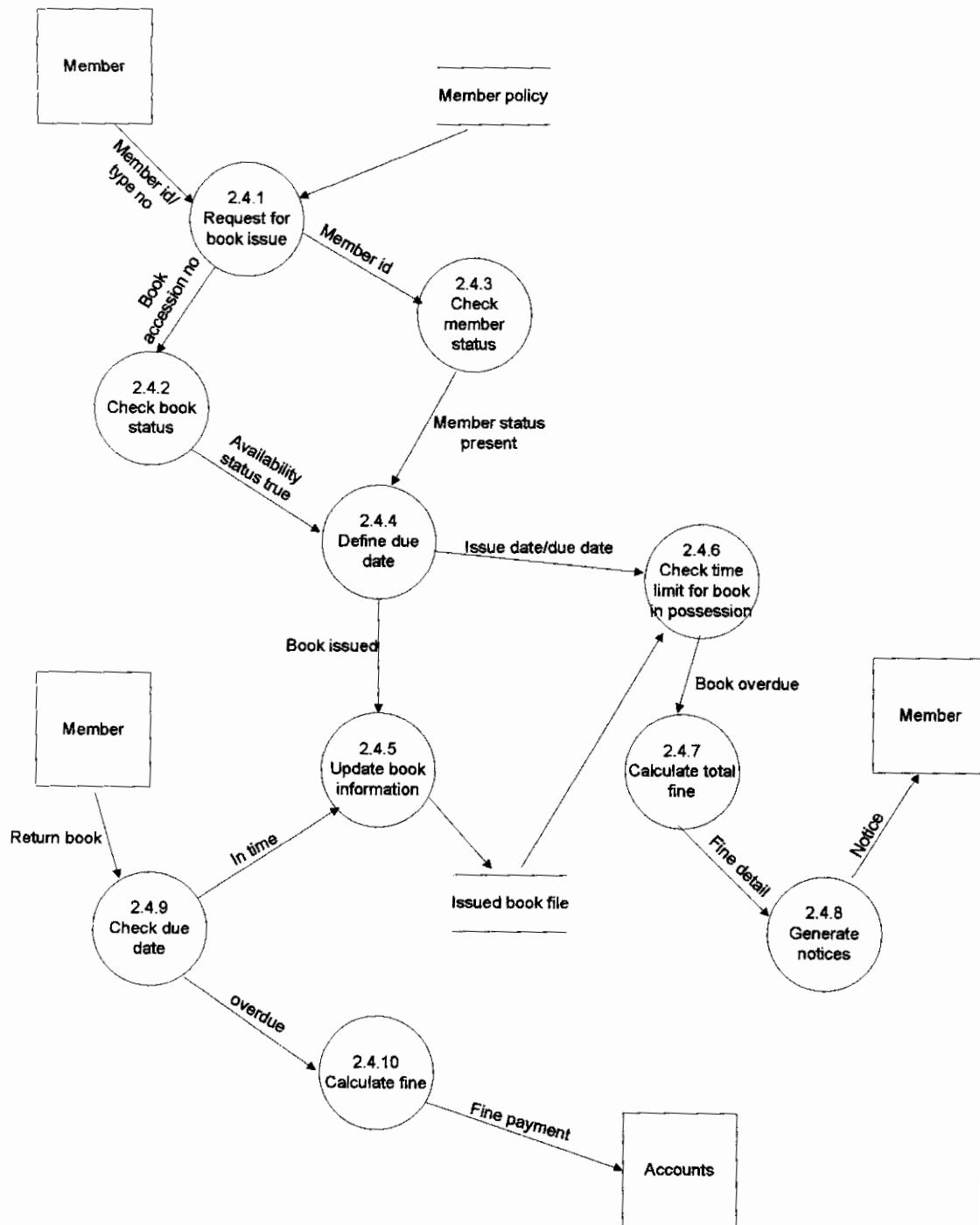
Level 2 DFD of Data Entry Processing



Level 2 DFD of Process Membership



Level 2 DFD of Circulation Desk Processing



BIBLIOGRAPHY AND REFERENCES

1. Library Automation: An introduction, Muhammad Riaz, Islamabad: EBSCO, 1991.
2. www.lib.berkley.edu
3. www.dpi.state.wi.us
4. Britannica Encyclopedia, 2002.
5. An Integrated Approach to Software Design, Pankaj Jalote, Published by N.K Mehra for Narosa Publishing House, 1998.
6. Software Engineering: A Practitioner's Approach, Fourth Edition, Roger S. Pressman, McGraw-Hill, 1997.
7. System Analysis and Design, Elias M. Awad, Galgotia Publications (P) Ltd., 2000-2001.
8. Developer/2000: Build Forms1, Student Guide Production 1.0 Volume 1 to 2 Oracle 8, Bulent Cinarkaya, Tushar Gadhia, Publishers: Kelly Lee, Renee Voss, January 1998.
9. Commercial Application Development using Oracle Developer/2000, Forms 5.0, Ivan Bayross, BPB Publications, 1999.
10. Oracle Education, Developer/2000: Build Reports, Volume 1. Student Guide, Louise Barnfield, Publishers: Jade Arrington, Geraldine Freitas, Kelley Lee, Kimberly Lee, January 1998.
11. <http://www.onwe.co.za>
12. www.dbasupport.com
13. www.geocities.com
14. www.stc.hill.af.mil