**LIBRARY MANAGEMENT SYSTEM**

**ABSTRACT**

The main objective of the Library Management system is organizing and managing the library tasks. Library is place where all kind of books are available. This is a web base application and only a registered user can access the application. Library Management system is developed to automate the task of entering the records of new book and retrieving the details of book available in the library. This system contains list of all the books. Using this system user can issue book to the library member, maintain their records, and can checks how many book are issued and how many books are available in the library. This system provides separate interface and login for librarian, students and faculties. Librarian can modify database. Using the library management system, user can also maintain the late fine of library member who return the issued book after the due date.

Users can search for books and renewal books. In the proposed system, we assume that each member will be having a identity card which can be used for the library book issue, fine payment etc. They can recommend for new books by just sending messages to the librarian from any where in the college. whenever library member wish to take a book, the book issued by the library authority will be check both the book details as well as the student details and store it in library database. They can view the issue and return dates of any book and due they have to pay

**1. INTRODUCTION**

In this project is an goal of to managing the library management system in school/ colleges or anywhere. In this software library administrator handling overall daily works and activities. Before using this project admin has register all the details like adding book details, creating author, and creating a student like that. Without this software we can’t managing the library. It’s very user friendly to use it. The project is aimed to develop by **JAVA** as Front end and **MS SQL SERVER** as Back end. The back end is used to store the information in this system.

**1.1 SYSTEM SPECIFICATION**

**1.1.1 HARDWARE SPECFICATION:**

* Processor : P 4 700 GHz.
* RAM : 4GB RAM
* Hard Disk Drive : 40 GB HDD

**1.1.2 SOFTWARE SPECIFICATION:**

* + Operating System : Windows XP/7/8/10
  + Front End : JAVA
  + Back End : MY SQL

1. **SYSTEM STUDY**

**2.1 EXISTING SYSTEM:**

In our existing system all the transaction of books are done manually, So taking

more time for a transaction like borrowing a book or returning a book and also for searching

of members and books. Another major disadvantage is that to preparing the list of books

borrowed and the available books in the library will take more time, currently it is doing as a

one day process for verifying all records. So after conducting the feasibility study we decided to make the manual Library management system to be computerized.

**2.1.1 DRAWBACKS:**

The existing system has the following drawbacks.

* Very hard to managing students
* Very hard managing book details manually.
* Difficult to calculate fine amount.

**2.2 PROPOSED SYSTEM:**

Proposed system is an automated Library Management System. Through our

software user can add members, add books, search members, search books, update

information, edit information, borrow and return books in quick time. Our proposed system

has the following advantages.

**2.2.1 FEATURES:**

* User friendly interface
* Fast access to database
* Less error
* More Storage Capacity
* Search facility
* Look and Feel Environment
* Quick transaction

1. **SYSTEM DESIGN AND DEVELOPMENT**

**3.1 FILE DESIGN**

The selection of the file system design approach is done according to the needs of the developers what are the needed requirements and specifications for the new design. It allowed us to identify where our proposal fitted in with relation to current and past file system development. Our experience with file system development is limited so the research served to identify the different techniques that can be used. The variety of file systems encountered show what an active area of research file system development is. The file systems may be from one of the two fundamental categories. In one category, the file system is developed in user space and runs as a user process. Another file system may be developed in the kernel space and runs as a privileged process. Another one is the mixed approach in which we can take the advantages of both aforesaid approaches. Each development option has its own pros and cons. In this article, these design approaches are discussed.

**3.2 INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:’

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

* Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
* It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.
* When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user
* will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**3.3 OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**3.4 DATABASE DESIGN**

Today's businesses depend on their databases to provide information essential for day-to-day operations, especially in case of electronic commerce businesses who has a definite advantage with up-to-date database access. Good design forms the foundation of any database, and experienced hands are required in the automation process to design for optimum and stable performance.

Software Solutions have been constantly working on these platforms and have attained a level of expertise. We apply proven methodologies to design, develop, integrate and implement database systems to attain its optimum level of performance and maximize security to meet the client's business model.

### Business needs addressed:

* Determine the basic objects about which the information is stored
* Determine the relationships between these groups of information and the objects
* Effectively manage data and create intelligent information
* Remote database administration or on site administrative support
* Database creation, management, and maintenance
* Information retrieval efficiency, remove data redundancy and ensure data security

**3.5 SYSTEM DEVELOPMENT**

**3.5.1 DESCRIPTION OF MODULES**

1. Admin Login
2. User Login
3. Add and update books
4. Search book
5. Calculate Fine

**User Registration**

**Admin login:**

 Admin is the one who administers the system by adding or removing e-books into and from the system respectively.

**User login:**

Students have to register themselves into the system to create an account. After registering successfully, they can then login into the system by entering 10 digit mobile number and their email id.

**Add and Update Books:**

The admin can add books to the system by entering the details of the books and can even update the details.

**Search option:**

Admin and Students can even search for books by entering the name of the book.View Order-The admin can view order for the books.The students can place order for the books and simultaneously the quantity of the book ordered will be decremented.

**Calculate Fine-** The student can view the issue and expiry date for the book issued and can even calculate fine.

1. **SYSTEM TESTING AND IMPLEMENTATION**

**SYSTEM TESTING**

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and application reside on a network and interoperate with many different operating system, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web application.

The distributed nature of client\server environments, the performance issues associated with transaction processing, the potential presence of a number of different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database and the requirements imposed on the server all combine to make testing of client\server architectures.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer based system. System testing is the state of implementation that is aimed at assuring that the system works accurately and efficiently. Testing is the vital to the success of the system. System testing makes the logical assumption that if all the parts of the system are correct, the goal will be successfully achieved.

**The objective of testing is as follows:**

* + Testing is the process of executing a program with the intent of finding an error.
  + A successful test is that one of the cover of undiscovered error.

### TESTING ISSUES

* Client GUI considerations
* Target environment and platform diversity considerations
* Distributed database considerations
* Distributed processing considerations

**TESTING METHODOLOGIES**

System testing is state of implementation, which is aimed at ensuring that the system works accurately and efficiently as expect before live operation commences. It certifies that the whole set of programs hang together.

System testing requires a test plan that consists of several key activities and step for run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system

Testing is the important stage in software development. the system test in implementation stage in software development process. The system testing implementation should be confirmation that all is correct and an opportunity to show the users that the system works as expected. It accounts the largest percentage of technical effort in the software development process.

Testing phase in the development cycle validates the code against the functional specification testing is vital to achievement of the system goals. The objective of the testing is to discover errors to fulfills this objective a series of test step unit, integration. validation and system tests were planned and executed the test steps are:

**System Testing**

Testing is an important phase in project development. System testing makes a logical assumption that if all parts of the system are correct, and the goal will be achieved successfully. The software must meet the user specification and it must satisfy according to the needs of the users.

Testing is the process of executing a project within the intend of finding errors. A good test case is one that has a high probability of finding an undiscovered error.

**Unit Testing**

Unit testing focuses verification efforts on the smallest unit of software design of the module. This is also known as “module testing”. This testing is carried out during programming stage itself. In this testing step, each module is found to be working satisfactorily as regards to the expected output of the modules.

**In Project**, Each module such customer registration module, request module, employee details module, stock module, vehicle module and area detail modules are tested individually for example, Customer details module can contain the more forms to maintain the information so all forms could be tested like entered information store appropriately in database access page or not. If correctly accessed means the testing of registration module successfully completed. Likewise all modules are tested successfully.

**Integration Testing**

Data can be lost across an interface, one module can have adverse effect on another sub function when combined it may not produce the desired major functions. Integration testing is a systematic testing for constructing test to uncover errors associated within an interface.

The objectives taken from unit tested modules and a program structure is built for integrated testing. All the modules are combined and the test is made.

A correction made in this testing is difficult because the vast expenses of the entire program complicated the isolation of causes. In this integration testing step, all the errors are corrected for next testing process.

**In Project,** Integration of two modules can be tested together such as customer registration details and customer login module for verification purposes providing proper accessibility to users. The communication of Registration and Login module can test and executed successfully.

**Validation Testing**

After the completion of the integrated testing, software is completely assembled as a package; interfacing error has been uncovered and corrected and a final series of software test validation begins.

Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software function in a manner that can be reasonably expected by the customer. After validation test has been conducted, one of two possible conditions exists:

**In this project,** Admin login details form Enter without username and password in textbox enter the submit button then Login failed message otherwise checks the both textbox value that is true means valid page displayed. Enter Password Displaying password character \*.if it displays the characters security is not availed so testing of software is failed.

**Output Testing**

The next process of validation testing, is output testing of the proposed system, since no system could be successful if it does not produce the required output in the specified format. Asking the user about the format required, list the output to be generated or displayed by the system under considerations.

Output testing is a different test whose primary purpose is to fully exercise the computer based system although each test has a different purpose all the work should verify that all system elements have been properly integrated and perform allocated functions.

The output format on the screen is found to be corrected as the format was designed in the system design phase according to the user needs for the hard copy also; the output testing has not resulted in any correction in the system.

**In project** All the forms are tested as it gives the necessary output to the user’s search such as view response details.

1. **CONCLUSION:**

This project has been used by both admin and student, it will be used by college or school anywhere. Student can take a book and returning a book very easily. It will know the library books and author details.

**FUTURE ENHANCEMENT**

A library management system is an very big concept, we just initializing the first step is for just creating books and separate student portal only, but our future enhancement is an student can took a book from online, its fully digitalized library, every student has some limitation to taken a book. Every one will have to like this application. This is our main concept to moving next level.

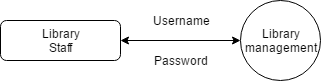
**BIBLIOGRAPHY**

* **Gea99b**  
  David M. Geary Graphic Java 2: Mastering the JFC, vol. II, Swing, third ed., Sun Microsystems Press, 1999.
* **CM96**  
  Peter Coad and Mark Mayfield, Java Design: Building Better Apps and Applets, Yourdon Press, 1996.
* **ELW98**  
  Robert Eckstein and Marc Loy and Dave Wood, Java Swing, O'Reilly, 1998.

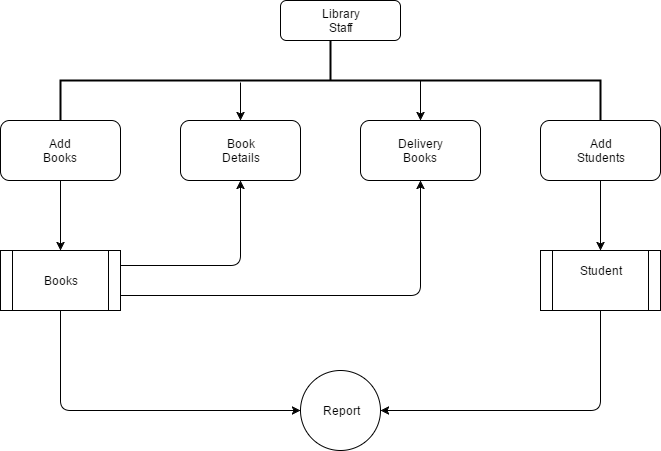
**APPENDICES**

1. **DATA FLOW DIAGRAM**

LEVEL 0:

****

LEVEL 1:



1. **TABLE STRUCTURE**

**TABLE NAME : ADMIN**

**PRIMARY\_KEY : ID**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int |  | Admin id |
| Username | Varchar | 30 | Admin username |
| password | Varchar | 30 | Admin password |

**TABLE NAME : STUDENT**

**PRIMARY KEY : student id**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int | 10 | Student id |
| Name | Varchar | 30 | Student name |
| Address | Varchar | 30 | Address |
| Contactno | Varchar | 10 | Contact number |
| Email id | Varchar | 10 | Email id |
| Password | Varchar | 10 | password |

**TABLE NAME : ISSUE BOOK**

**PRIMARY KEY : id**

**FOREIGN KEY : bid,sid**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPE** | **SIZE** | **DESCRIPTION** |
| Id | Int | 10 | Issue book id |
| Bid | Int | 10 | Book id |
| Sid | Int | 10 | Student id |
| Issue date | Varchar | 30 | Issue date |
| Return date | Varchar | 30 | Return date |
| Fine | Int | 10 | fine |

1. **SAMPLE CODING**

<?

session\_start();

error\_reporting(0);

\_x0012\_

if($\_SESSION['alogin']!=''){

$\_SESSION['alogin']='';

}

if(isset($\_POST['login']))

{

//code for captach verification

if ($\_POST["vercode"] != $\_SESSION["vercode"] OR $\_SESSION["vercode"]=='') {

echo "<script>alert('Incorrect verification code');</script>" ;

}

else {

$username=$\_POST['username'];

$password=$\_POST['password'];

$sql ="SELECT UserName,Password FROM admin WHERE UserName=:username and Password=:password";

$query= $dbh -> prepare($sql);

$query-> bindParam(':username', $username, PDO::PARAM\_STR);

$query-> bindParam(':password', \_x0012\_$password, PDO::PARAM\_STR);

$query-> execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

echo "<script>alert('$sql');</script>";

if($query->rowCount() > 0)

{

$\_SESSION['alogin']=$\_POST['username'];

echo "<script type='text/javascript'> document.location ='admin/dashboard. '; </script>";

} else{

echo "<script>alert('Invalid Details');</script>";

}

}

}

?>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Library Management System</title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

</head>

<body>

<!------MENU SECTION START-->

<? include('includes/header. ');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">ADMIN LOGIN FORM</h4>

</div>

</div>

<!--LOGIN PANEL START-->

<div class="row">

<div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-3" >

<div class="panel panel-info">

<div class="panel-heading">

LOGIN FORM

</div>

<div class="panel-body">

<form role="form" method="post">

<div class="form-group">

<label>Enter Username</label>

<input class="form-control" type="text" name="username" autocomplete="off" required />

</div>

<div class="form-group">

<label>Password</label>

<input class="form-control" type="password" name="password" autocomplete="off" required />

</div>

<div class="form-group">

<label>Verification code : </label>

<input type="text" name="vercode" maxlength="5" autocomplete="off" required style="width: 150px; height: 25px;" />&nbsp;<img src="captcha. ">

</div>

<button type="submit" name="login" class="btn btn-info">LOGIN </button>

</form>

</div>

</div>

</div>

</div>

<!---LOGIN PABNEL END-->

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<? include('includes/footer. ');?>

<!-- FOOTER SECTION END-->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</script>

</body>

</html>

<?

session\_start();

error\_reporting(0);

include('includes/config. ');

if(strlen($\_SESSION['login'])==0)

{

header('location:index. ');

}

else{

if(isset($\_GET['del']))

{

$id=$\_GET['del'];

$sql = "delete from tblbooks WHERE id=:id";

$query = $dbh->prepare($sql);

$query -> bindParam(':id',$id, PDO::PARAM\_STR);

$query -> execute();

$\_SESSION['delmsg']="Category deleted scuccessfully ";

header('location:manage-books. ');

}

?>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />

<meta name="description" content="" />

<meta name="author" content="" />

<title>Online Library Management System | Manage Issued Books</title>

<!-- BOOTSTRAP CORE STYLE -->

<link href="assets/css/bootstrap.css" rel="stylesheet" />

<!-- FONT AWESOME STYLE -->

<link href="assets/css/font-awesome.css" rel="stylesheet" />

<!-- DATATABLE STYLE -->

<link href="assets/js/dataTables/dataTables.bootstrap.css" rel="stylesheet" />

<!-- CUSTOM STYLE -->

<link href="assets/css/style.css" rel="stylesheet" />

<!-- GOOGLE FONT -->

<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />

</head>

<body>

<!------MENU SECTION START-->

<? include('includes/header. ');?>

<!-- MENU SECTION END-->

<div class="content-wrapper">

<div class="container">

<div class="row pad-botm">

<div class="col-md-12">

<h4 class="header-line">Manage Issued Books</h4>

</div>

<div class="row">

<div class="col-md-12">

<!-- Advanced Tables -->

<div class="panel panel-default">

<div class="panel-heading">

Issued Books

</div>

<div class="panel-body">

<div class="table-responsive">

<table class="table table-striped table-bordered table-hover" id="dataTables-example">

<thead>

<tr>

<th>#</th>

<th>Book Name</th>

<th>ISBN </th>

<th>Issued Date</th>

<th>Return Date</th>

<th>Fine in(USD)</th>

</tr>

</thead>

<tbody>

<?

$sid=$\_SESSION['stdid'];

$sql="SELECT tblbooks.BookName,tblbooks.ISBNNumber,tblissuedbookdetails.IssuesDate,tblissuedbookdetails.ReturnDate,tblissuedbookdetails.id as rid,tblissuedbookdetails.fine from tblissuedbookdetails join tblstudents on tblstudents.StudentId=tblissuedbookdetails.StudentId join tblbooks on tblbooks.id=tblissuedbookdetails.BookId where tblstudents.StudentId=:sid order by tblissuedbookdetails.id desc";

$query = $dbh -> prepare($sql);

$query-> bindParam(':sid', $sid, PDO::PARAM\_STR);

$query->execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query->rowCount() > 0)

{

foreach($results as $result)

{ ?>

<tr class="odd gradeX">

<td class="center"><? echo htmlentities($cnt);?></td>

<td class="center"><? echo htmlentities($result->BookName);?></td>

<td class="center"><? echo htmlentities($result->ISBNNumber);?></td>

<td class="center"><? echo htmlentities($result->IssuesDate);?></td>

<td class="center"><? if($result->ReturnDate=="")

{?>

<span style="color:red">

<? echo htmlentities("Not Return Yet"); ?>

</span>

<? } else {

echo htmlentities($result->ReturnDate);

}

?></td>

<td class="center"><? echo htmlentities($result->fine);?></td>

</tr>

<? $cnt=$cnt+1;}} ?>

</tbody>

</table>

</div>

</div>

</div>

<!--End Advanced Tables -->

</div>

</div>

</div>

</div>

</div>

<!-- CONTENT-WRAPPER SECTION END-->

<? include('includes/footer. ');?>

<!-- FOOTER SECTION END-->

<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->

<!-- CORE JQUERY -->

<script src="assets/js/jquery-1.10.2.js"></script>

<!-- BOOTSTRAP SCRIPTS -->

<script src="assets/js/bootstrap.js"></script>

<!-- DATATABLE SCRIPTS -->

<script src="assets/js/dataTables/jquery.dataTables.js"></script>

<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>

<!-- CUSTOM SCRIPTS -->

<script src="assets/js/custom.js"></script>

</body>

</html>

<? } ?>