|  |  |  |
| --- | --- | --- |
| D:\Aptech - FPT\semester 2\eProject\documentation\LogoAptech.png | D:\Aptech - FPT\semester 2\eProject\documentation\LogoAptech.png | D:\Aptech - FPT\semester 2\eProject\documentation\LogoAptech.png |

Center Name: ACE-HCMC-2-FPT (VN.033).

Address: 590 Cach Mang Thang 8 Street, District 3, Ho Chi Minh City, Vietnam.



**LIBRARY MANAGEMENT SYSTEM**

**Design Document**

|  |  |  |  |
| --- | --- | --- | --- |
| **Faculty:** |  | **Ms. Dao Ngoc Anh** |  |
| **Batch No:** |  | **F2131405T0** |  |
| **Group No.:** |  | **3** |  |
| **Students:** |  | **Fullname** | **Roll No.** |
|  | 1. | Hua Tran Huu Trung | Student812912 |
|  | 2. | Tra Phuc Vinh Uy | Student800173 |
|  | 3. | To Thanh Dat | Student813016 |

**Dec, 2014**

# 

# REVIEW 1

1. **Introduction**

The main objective of the Library Management system project is discipline of the planning, organizing and managing the library tasks. Our project aims at making the task of library easy.

Library Management is entering the records of new book and retrieving the details of book available in the library. We can issue book to the library member and maintain their records and can also checks how many book are issued and stock available in the library. In the project we can maintain the late fine of library member who return the issued book after the due date.

1. **Existing Scenario**

* Currently in library, all records are managed on paper. It is hardly for Staffs to manage them when somebody issues or returns the books, how many total of books in each category, etc.
* Each of the staffs of separate department handle the different duties in library such as Book account: manage the books in library, IR account: manage issuing/returning the books. Member account: manage members of library. Accounting account: manage fine fee. Admin account: manage staff account and all functions in system.
* Managers have to maintain list of books along with details such as author name, book’s title, book’s category, etc. When the books are to be issued, they have to manually search for that book and again for issuing book they have to maintain a register which will have details of book issued such as book’s title, staff’s detail, date of issue, date of return, etc.

1. **Proposed Solution**

Proposed system is an automated Library Management System. Through our software, user can add members, add books or search books, search members with multi conditionals or edit member’s information, edit book’s information, borrow and return books in quick time. Our proposed system has the following advantages:

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

All the manual difficulties in managing the Library have been rectified by implementing computerization.

1. **Function Of Application**
2. Login/Logout: allow Users (Staffs in Library) login/logout into System. For each level of user, there will be the respective rights.
3. Change Password: allow Users can change their Password.
4. Management Member’s account: allow Member account manages member’s information.
5. Management Staff’s account: allow Admin account manages Staff’s information.
6. Management Book: allow Book account manages books in library.
7. Management Category: allow Admin account manages categories of Book.
8. Issue Book: allow IR account can issues books for member who has requested if those books are available.
9. Return Book: allow IR account can accept returned book from member. And check due date of that book is legal or not. If it is not legal, member must pay fee fine depend how many days pass over.
10. Accept fine: allow Accounting account can accept fine fee from member, and update status member paid fee for library, and can continue for being issued books later.
11. Generate report for Book account: this will provide how many books depend on their category and total books which the Library has.
12. Generate report for Member account: This will provide the details of issuing member as member name, book’s title, book’s id, issue date, return date.
13. Generate report for IR account: This will provide the details of issuing/returning book as book’s ISBN, book’s title, total issue, total return.
14. Generate report for Accounting account: This will provide the details of paying member as member name, book’ title, book’s id, issue date, return date, amount day later, amount fine fee, fine status.
15. Search: The user will be able to perform search multi conditionals based on member’s name, book’s id, book’s title, issue date, returning date, category....
16. **The Roles of Application**
17. **All member:**

* Login.
* Change Password.

1. **Admin Account:**

* Staff Management (View, Insert, Update).
* Category’s book Management (View, Insert, Update, Lock).
* Create/View all report.
* View all other functions
* Search by all.

1. **Book Account:**

* Book Management (View, Insert, Update, Lock).
* Create/View Category-Book report.
* Search by Book’s Id, title, category…

1. **Member Account:**

* Member Management (View, Insert, Update, Lock).
* Create/View Member-Issue report.
* Search by Member’s Id, name.

1. **IR Account:**

* Issue Management (View, Issue, Update).
* Return Management (View, Return, Update).
* Create/View Book-Issue-Return report.
* Search by Member’s Id, Book’s Id, Issue’s date, Issue’s Id.

1. **Accounting Account:**

* Fine Management (View, Update)
* Create/View Member-Issue/Return-Fine report.
* Search by Member’s Id, name, Issue’s date, Issue’s Id.

1. **Function Diagram**
2. **Admin function**



1. **Book Account function**



1. **Member Account function**



1. **IR Account function**



1. **Accounting Account function**



1. **Hardware/Software Requirements**
2. **Hardware**

* A minimum computer system that will help you access all the tools in the courses is a Pentium 4 or better
* 512 MB of RAM or better
* HDD 10G free space available or better

1. **Software**

* Operating system: Windows XP (or higher if possible)
* Netbeans 8.0.1
* JDK 1.7
* SQL Server 2012
* JDBC Driver 4

1. **Task sheet**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Ref.No: 3** | | **Project Title:**  **Library Management System** | **Date of Preparation of Activity Plan** | | | |
| **Sr. No.** | **Task** | | **Actual Start Date** | **Actual Days** | **Team Member Names** | **Status** |
| 01 | Introduction | | Dec, 22, 2014 | 4 | All Members | Completed |
| 02 | Exiting Scenario | | All Members | Completed |
| 03 | Proposed Solution | | All Members | Completed |
| 04 | Function Of Application and The Roles of Application | | All Members | Completed |
| 05 | Function Diagram | | All Members | Completed |
| 06 | Hardware/Software Requirements | | All Members | Completed |
| 07 | Task sheet | | All Members | Completed |

|  |  |  |
| --- | --- | --- |
|  | Prepare By: Group 3 | Approved By: Faculty |
| Date: Dec, 2014 | Team Leader  **Hua Tran Huu Trung** | **Dao Ngoc Anh** |

# 

# REVIEW 2

**Architecture and Design of Project**

Tier 1

Presentation

Tier 3

Data Access

Tier 2

Business Logic

Three-Tier Architecture of the Project

1. **Data Flow Diagram**

**Definition:** Data Flows Diagram (DFD) describes the information flow in the system. The next step of system analysis is to consider in detail the information necessary for the implementation for functions discussed above and the one necessary for the improvement of the functions. Modeling tool frequently used for this purpose is DFD. DFD will support 4 main activities:

* **Analysis:** DFD is used to determine requirement of users.
* **Design:** DFD is used to map out plan and illustrate solution to analysis and users while designing a new system.
* **Communication:** one of the strength of DFD is its simplicity and ease to understand to analysts and users;

**Document:** DFD is used to provide special description of requirement and system design. DFD provide an overview of key functional components of the system but it does not provide any detail on these components. We have to use other tools like database dictionary, process specification to get an idea of which information will be exchanged and how.

**The main components of Context Diagram:**

**External**

**The process**: Show the common function of system.

**The external factors**: External factors can be a person, a group of persons or an organization that are sources of information for the systems and are where system products are transferred to.

**The data flow**: Describe the movement of information from one part of the system to another.

**The data store**: The Data Store is used to model a collection of data packets at rest. A store is represented graphically by two parallel lines. The name of a Data Store that identifies the store is the plural of the name of the packets that are carried by flows into and out of the Data Store.

Data Flow

External

Process

Data Store

**DFD level 0 – Context Diagram**



**DFD – Login**



**DFD – Admin**













1. **Database Design**

**Entity Relationship Diagram (ERD)**



**Entity and Properties**

1. Members



1. Staffs



1. Categories



1. Books



1. Copies



1. IRBooks



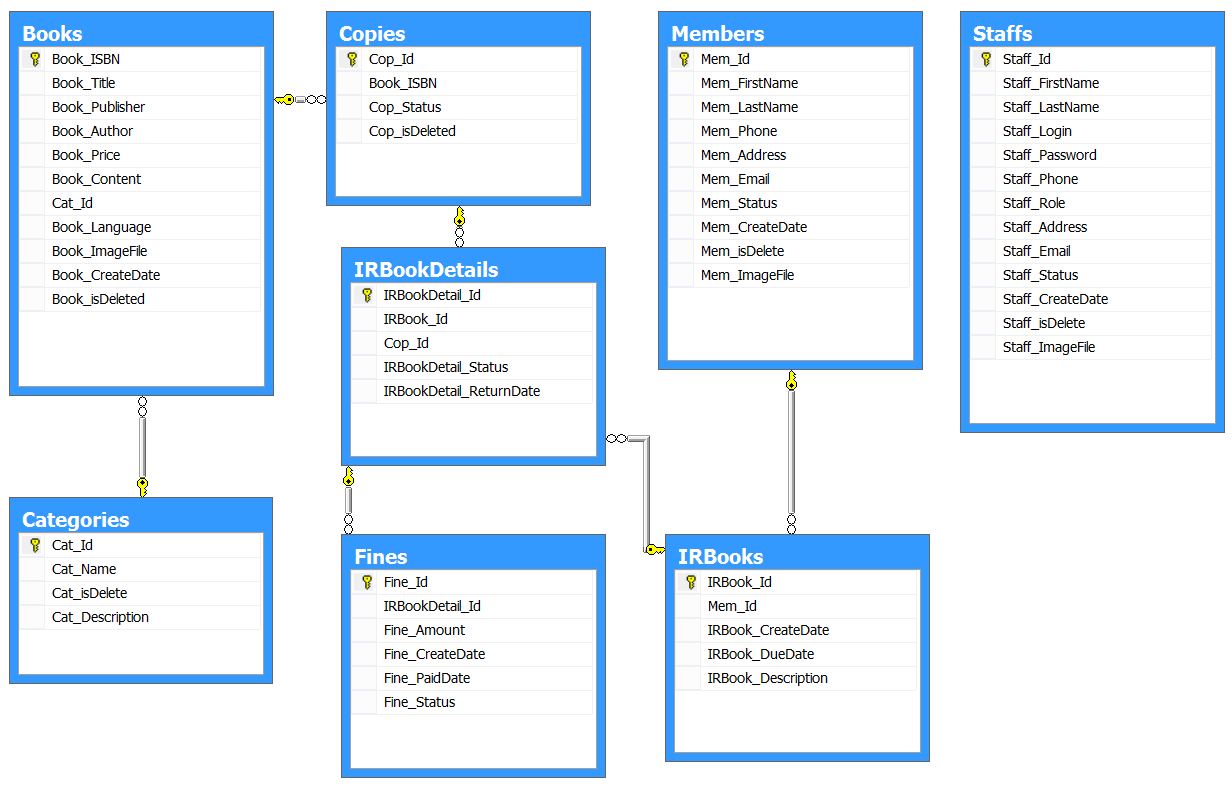
1. IRBookDetails



1. Fines



**Database Relationship Diagram (DRD)**



**Database Structure**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Members** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Mem\_Id | char(36) | No | newid() | Primary |  | ID of Member |
| Mem\_FirstName | nvarchar(30) | No |  |  |  | FirstName of Member |
| Mem\_LastName | nvarchar(50) | No |  |  |  | LastName of Member |
| Mem\_Phone | varchar(20) |  |  |  |  | Phone of Member |
| Mem\_Address | nvarchar(200) |  |  |  |  | Address of Member |
| Mem\_Email | varchar(50) | No |  |  | Unique | Email of Member |
| Mem\_Status | bit | No | 1 |  |  | Status of Member |
| Mem\_CreateDate | datetime | No | getdate() |  |  | Create date of Member |
| Mem\_isDelete | bit | No | 0 |  |  | Record status is deleted or not |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Roles** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Role\_Id | varchar(15) | No |  | Primary |  | ID of Role |
| Role\_Name | nvarchar(20) | No |  |  | Unique | Name of Role |
| Role\_Description | nvarchar(100) |  |  |  |  | Description of Role |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Staffs** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Staff\_Id | Char(36) | No | newid() | Primary |  | ID of Staff |
| Staff\_FirstName | nvarchar(30) | No |  |  |  | First name of Staff |
| Staff\_LastName | nvarchar(50) | No |  |  |  | Last name of Staff |
| Staff\_Login | varchar(25) | No |  |  | Unique | Login name of Staff |
| Staff\_Password | varchar(50) | No |  |  |  | Password of Staff |
| Staff\_Phone | varchar(20) |  |  |  |  | Phone of Staff |
| Role\_Id | varchar(15) | No |  | Foreign |  | ID of Role |
| Staff\_Address | nvarchar(200) |  |  |  |  | Address of Staff |
| Staff\_Email | varchar(50) | No |  |  | Unique | Email of Staff |
| Staff\_Status | bit | No | 1 |  |  | Status of Staff (0: Banned,1:Active) |
| Staff\_CreateDate | datetime | No | getdate() |  |  | Create date of Staff |
| Staff\_isDelete | bit | No | 0 |  |  | Record status is deleted or not |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Categories** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Cat\_Id | char(36) | No | newid() | Primary |  | ID of Category |
| Cat\_Name | nvarchar(30) | No |  |  | Unique | Name of Category |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Languages** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Lang\_Id | varchar(7) | No |  | Primary |  | ID of Language(depend ISO 639-2 Code) |
| Lang\_Name | nvarchar(30) | No |  |  | Unique | Name of Language |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Books** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Book\_ISBN | varchar(13) | No |  | Primary |  | ISBN of Book |
| Book\_Title | nvarchar(50) | No |  |  |  | Title of Book |
| Book\_Publisher | nvarchar(50) |  |  |  |  | Publisher of Book |
| Book\_Author | nvarchar(50) |  |  |  |  | Author of Book |
| Book\_Price | money | No |  |  |  | Price of Book |
| Cat\_Id | char(36) | No |  | Foreign |  | ID of Category |
| Lang\_Id | varchar(7) | No |  | Foreign |  | ID of Language |
| Book\_ImageFile | varchar(255) | No | /imgBook/nocover.png |  |  | Path file image of Book |
| Book\_CreateDate | datetime | No | getdate() |  |  | Create date of Book |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Copies** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Cop\_Id | char(36) | No |  | Primary |  | ID of Copy |
| Book\_ISBN | varchar(13) | No |  | Foreign |  | ISBN of Book |
| Cop\_Status | bit | No | 1 |  |  | Status of Copy |
| Cop\_isDeleted | bit | No | 0 |  |  | Record status is deleted or not |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Orders** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Order\_Id | char(36) | No | newid() | Primary |  | ID of Order |
| Mem\_Id | char(36) | No |  | Foreign |  | ID of Member |
| Order\_CreateDate | datetime | No | getdate() |  |  | Create date of Order |
| Order\_DueDate | datetime | No |  |  |  | Due date of Order |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **OrderDetails** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| OrderDetail\_Id | char(36) | No | newid() | Primary |  | ID of OrderDetail |
| Order\_Id | char(36) | No |  | Foreign |  | ID of Order |
| Cop\_Id | varchar(16) | No |  | Foreign |  | ID of Copy |
| OrderDetail\_Status | bit | No | 0 |  |  | Status of OrderDetail |
| OrderDetail\_ReturnDate | datetime |  |  |  |  | Return date of OrderDetail |
| OrderDetail\_isDeleted | bit | No | 0 |  |  | Record status is deleted or not |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Fines** | | | | | | |
| **Column Name** | **Data Type** | **Allow Null** | **Default Value** | **Key** | **Validation** | **Description** |
| Fine\_Id | char(36) | No | newid() | Primary |  | ID of Fine |
| OrderDetail\_Id | char(36) | No |  | Foreign |  | ID of OrderDetail |
| Fine\_Amount | money | No |  |  | >=0 | Amount money of Fine |
| Fine\_CreateDate | datetime | No | getdate() |  |  | Create of Fine |
| Fine\_PaidDate | datetime |  |  |  |  | Paid date of Fine |
| Fine\_Status | bit | No | 0 |  |  | Status of Fine |

1. **Algorithms**

* **Issue Book**



* **Return Book and Paid Fine**



1. **Task Sheet**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Ref.No: 3** | | **Project Title:**  **Library Management System** | **Date of Preparation of Activity Plan** | | | |
| **Sr. No.** | **Task** | | **Actual Start Date** | **Actual Days** | **Team Member Names** | **Status** |
| 01 | Data Flow Diagram | | Dec, 23, 2014 | 5 | All Members | Completed |
| 02 | Entity Relationship Diagram (ERD) | | All Members | Completed |
| 03 | Entity and Properties | | All Members | Completed |
| 04 | Database Relationship Diagram (DRD) | | All Members | Completed |
| 05 | Database Structure | | All Members | Completed |
| 06 | Algorithms | | All Members | Completed |
| 07 | Task sheet | | All Members | Completed |

|  |  |  |
| --- | --- | --- |
|  | Prepare By: Group 3 | Approved By: Faculty |
| Date: Dec, 2014 | Team Leader  **Hua Tran Huu Trung** | **Dao Ngoc Anh** |