INDEX

|  |  |  |
| --- | --- | --- |
| **NUMBER** | **NAME** | **PAGE NUMBER** |
| 1 | OVERVIEW |  |
| 2 | REQUIREMENT |  |
| 3 | ADVANTAGES |  |
| 4 | ER DIAGRAM |  |
| 5 | NORMALIZATION |  |
| 6 | SCHEMA |  |

**OVERVIEW**

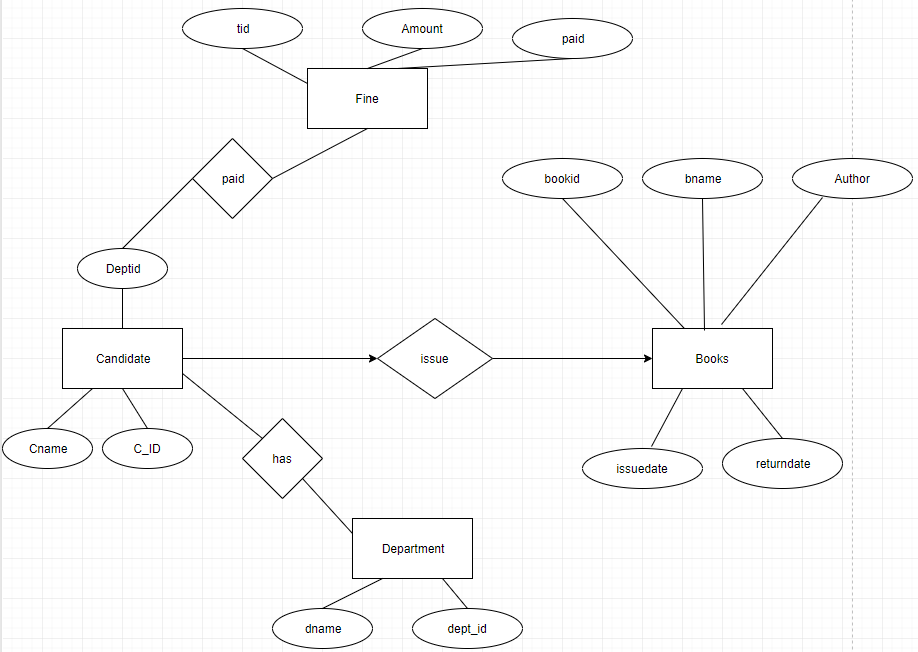
Library Management System aims to remove the paper based way of storing data by storing the data in databases on computers. Another objective that this project aims to accomplish is not to let students search for books in library, but instead they should do it on the website.

Library Management System is a website which provides a simple interface for college students to look up for any book they need, and make request for that book, can see in whose possession a particular book was the last time, can see how many books the user has issued with the request date, issue date, and return date. Fine can be calculated in case any user returns the book after the return date is due.

**ADVANTAGES**

* As whole system is online paper work will be reduced, human effort will be less.
* Data Redundancy will be reduced.
* Inconsistency will be less.
* User will be able to access data easily as whole system is centralize.
* Data will be shared among various users
* Atomicity: - Atomicity means if a task or changes has to be happened in Database, then that task will be completed 100% or won’t complete at all, before this task or changes are to be happened old values will be stored so that if task fails old values will be restored and consistency will be maintained.
* Security: -First time we need to register and in next semester need to register again and submit our id-card incase of offline, in our system user needs to register for once and each data will be saved.
* Concurrent Access: - In concurrent access multiple candidates can access system simultaneously, staff will be able to manage multiple users easily.

**ER DIAGRAM**



**Normalization**

We can apply 1nf to the database.

On the book record table, 1nf apply for a multivalued attribute in this table.

Like book name is a multivalued attribute. Our library management project. Also candidate table we can apply 1nf. like attribute divide in (f\_name,L\_name).

**SCHEMA**

DATABASE TABLES :

1. CANDIDATE :

CID – PRIMARY KEY

|  |  |  |  |
| --- | --- | --- | --- |
| CANDIDATE | | | |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DESCRIPTION |
| CID | VARCHAR(4) | PRIMARY KEY | ID NUMBER OF THE STUDENT |
| CNAME | VARCHAR(30) | NOT NULL | NAME OF THE STUDENT |
| TYPE | VARCHAR(7) | NOT NULL | TYPE OF THE STUDENT LIKE REGULAR OR DETAINED |
| DEPTID | VARCHAR(4) | FOREIGNKEY | DEPARTMENT ID OF THE CANDIDATE |

2. ISSUE

TID – PRIMARY KEY

BOOKID- FOREIGNKEY

OWNERID – FOREIGNKEY

|  |  |  |  |
| --- | --- | --- | --- |
| ISSUE | | | |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DESCRIPTION |
| TID | INT(11) | PRIMARY KEY | TRANSACTION ID OF FINE PAID |
| BOOKID | VARCHAR(8) | FOREIGNKEY | UNIQUE IDENTIFIER FOR EACH BOOK UNDER COMMON ID |
| OWNERID | VARCHAR(4) | FOREIGNKEY | UNIQUE IDENTIFIER FOR EACH CANDIDATE UNDER COMMON ID |
| ISSUE\_DATE | DATETIME | NOT NULL | DATE WHEN BOOK WAS ISSUED |
| RETURN\_DATE | DATETIME | NOT NULL | DATE WHEN BOOK WAS RETURNED |
| RETURNED\_STATUS | TINYINT(1) | NOT NULL | STATUS OF BOOK RETURNED OR NOT |

3.FINE

TID – FOREIGNKEY

|  |  |  |  |
| --- | --- | --- | --- |
| FINE | | | |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DESCRIPTION |
| TID | INT(4) | FOREIGNKEY | UNIQUE IDENTIFIER FOR EACH TRANSACTION UNDER COMMON ID |
| FINE\_AMOUNT | INT(10) | NONE | AMOUNT TO BE PAID BY THE CANDIDATE |
| PAID | TINYINT(1) | NOTNULL | IF AMOUNT WILL PAID THEN YES OTHERWISE NO |

4.BOOKS:

BOOKID – PRIMARY KEY

|  |  |  |  |
| --- | --- | --- | --- |
| BOOKS | | | |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DESCRIPTION |
| BOOKID | VARCHAR(5) | PRIMARY KEY | ID NUMBER OF THE BOOK |
| BOOKNAME | VARCHAR(100) | NOT NULL | NAME OF THE BOOK |
| AUTHOR | VARCHAR(50) | NONE | NAME OF THE AUTHOR OF THE BOOK |

5.DEPARTMENT

DEPTID – PRIMARY KEY

|  |  |  |  |
| --- | --- | --- | --- |
| DEPARTMENT | | | |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DESCRIPTION |
| DEPTID | VARCHAR(4) | PRIMARY KEY | ID NUMBER OF THE DEPARTMENT |
| DNAME | VARCHAR(30) | NOT NULL | NAME OF THE DEPARTMENT |

6.BOOKSID

COPYID – PRIMARY KEY

|  |  |  |  |
| --- | --- | --- | --- |
| DEPARTMENT | | | |
| ATTRIBUTE | DATATYPE | CONSTRAINT | DESCRIPTION |
| COPYID | VARCHAR(8) | PRIMARY KEY | ID NUMBER OF THE DEPARTMENT |
| BOOKID | VARCHAR(5) | NOT NULL | NAME OF THE DEPARTMENT |
| STATUS | TINYINT(1) | NOT NULL | STATUS IF BOOK IS ISSUED OR NOT |