

IT332 – Advanced Database Project

A University Library Database



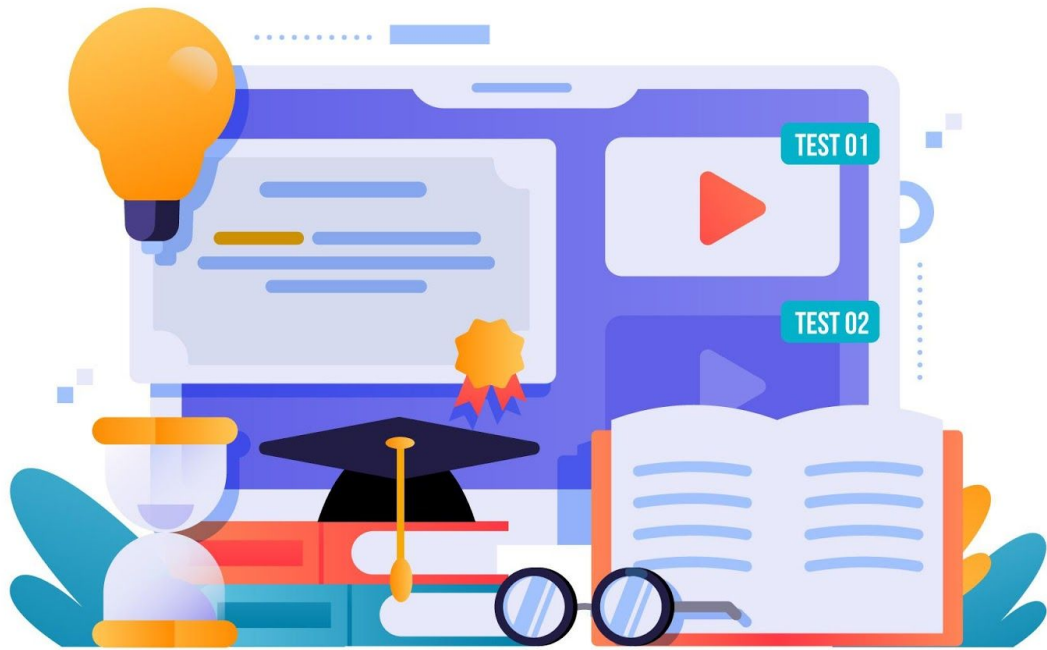
Fai Alsalehi - 381210927

Rahaf Alturki - 381213284

Hind Alzaidan - 381213135

Renad Alsweed - 381210443

QU LIBRARY



Our library provides materials in varied formats to meet the needs of students, as well as employees. It also provides many services that make the life of a student easier and much more organized.

1.REQUIREMENTS DOCUMENT

A. DATA REQUIREMENTS

1- Book: Books in our library are classified based on the **author, category, type, book name (book_name) and book ID (book_id)**.

2- Members: Members get to loan books, book a studying room with varying sizes and durations. The library keeps records of Members' ID's, Passwords, Fullnames, Emails, Date of Birth, and finally member type (**member_id, password, name, email, member_type_DOB**).

3- Privileges: Members are given different privileges based on their privilege ID, which will then allow them to get longer Loan periods, along with different room sizes and durations. The attributes included in this table are: **Loan_period, Privilege_ID and Book_duration**.

4- Loan: Our library gives members the chance to loan books and/ or any sort of media they need. The 'Loan' table has the following attributes: (Loan_number, Due_date), which makes it easier to identify members who were loaned.

5- Fine: Members who do not return books on time or return damaged items will have to pay a fine, and the 'Fine' table includes **date, state, description and amount** of the fine members must pay.

6- Staff : Staff are the people that work in the library to make sure everything is organized and the items' information are accurate. They also manage members' records and monitor loan history. The staff table in our database contains three main attributes, which are (**Phone, Email and S_ID**).

7- Room: The library has a lot of rooms, from studying rooms to meeting rooms, and they are all available for use for a specified duration. The library stores information about the **size** and the number of the room (**Room_number**).

8- Other Services: This table refers to services such as printing and copying, using computers with a working internet connection. It includes **description**, and the name of the service (**service_name**).

9- Branch: The branch table specifies which University branch it is based on its Phone number (**phone**), Location (**location**), Branch ID(**b_id**) and Name (**name**).

B. TRANSACTION REQUIREMENTS

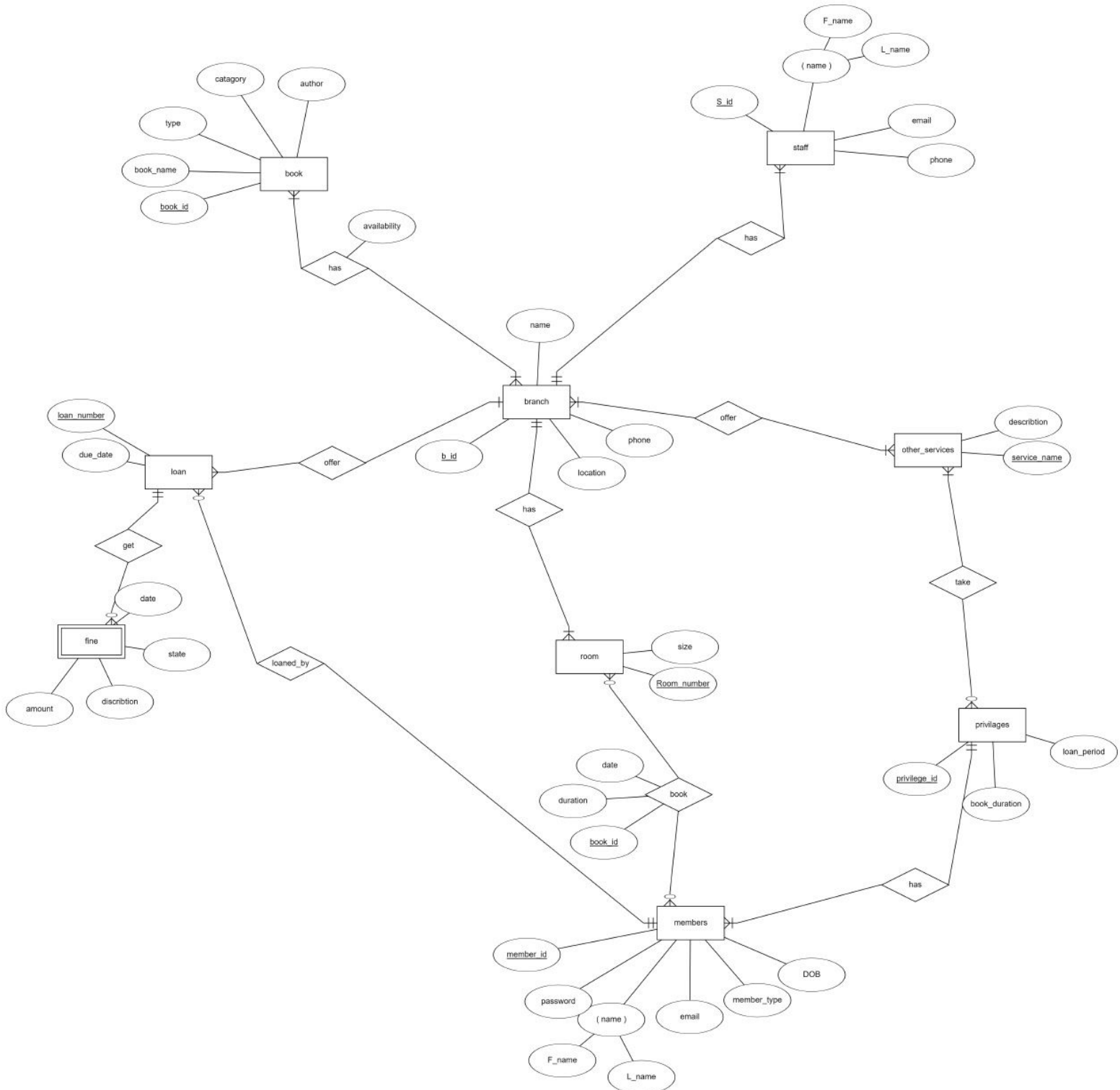
1. Data Manipulation

- (INSERT – DELETE – UPDATE) data of **Book**.
- (INSERT – DELETE – UPDATE) data of **Members**.
- (INSERT – DELETE – UPDATE) data of **Room**.
- (INSERT – DELETE – UPDATE) data of **Staff**.
- (INSERT – DELETE – UPDATE) data of **Fine**.
- (INSERT – DELETE – UPDATE) data of **Loan**.
- (INSERT – DELETE – UPDATE) data of **Privileges**.
- (INSERT – DELETE – UPDATE) data of **Branch**.
- (INSERT – DELETE – UPDATE) data of **Other Services**.

2. Queries

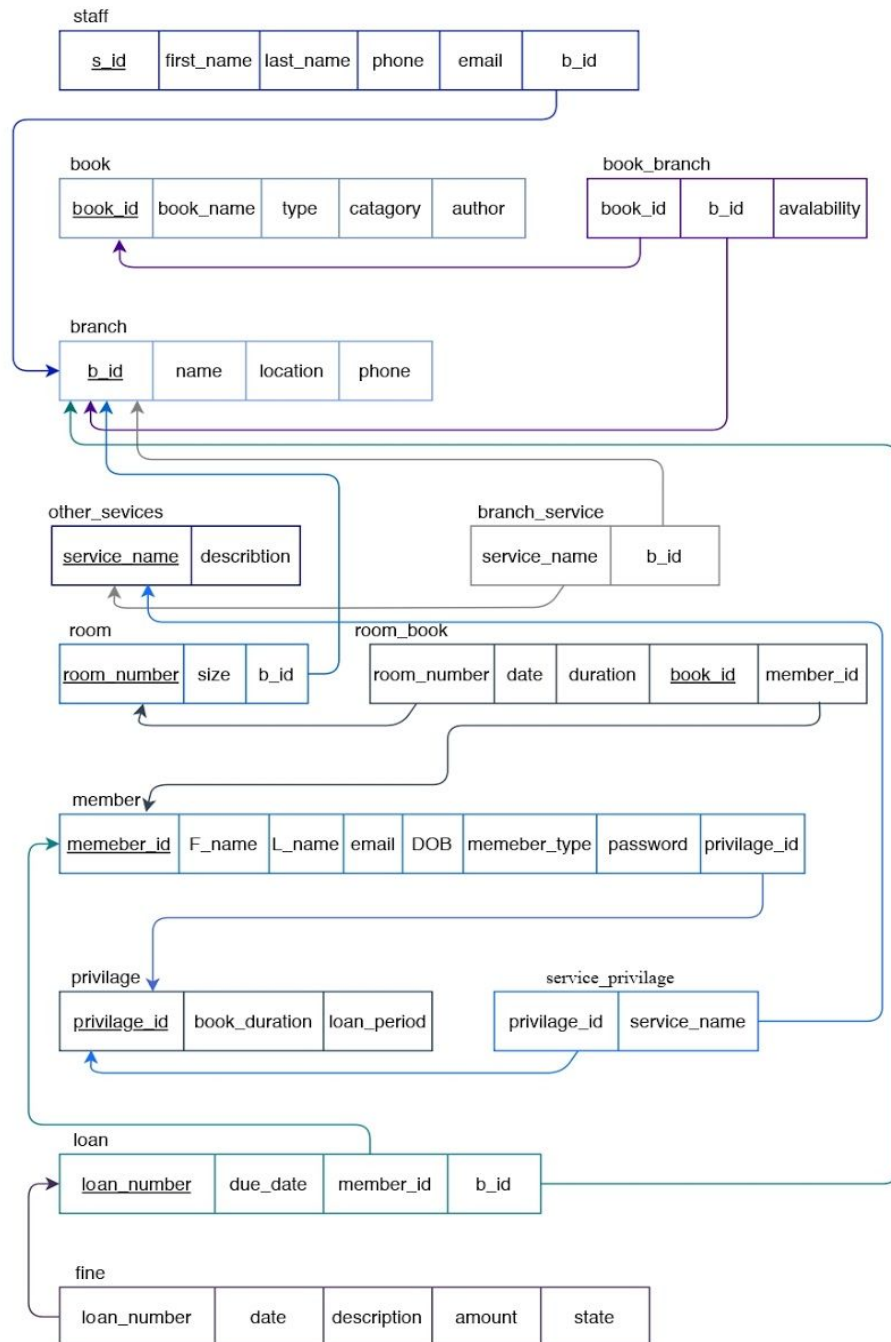
- list of all book sections in the library.
- List of the book_id and book name.
- A list of all members with their privileges.
- List of room size and times.
- List of all services available in the library
- List b_id of all branches.
- List S_id of all staff.
- List all pending fines that are way past their due date.

2. UML /ER MODELS

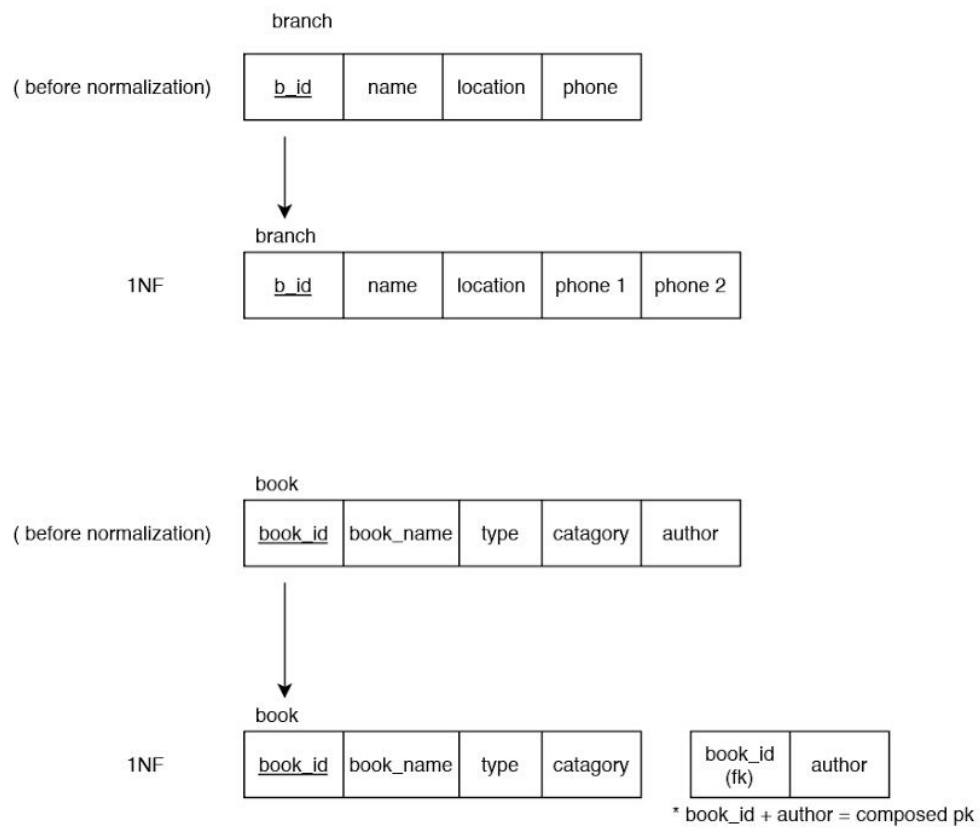


3. RELATIONAL MAPPING & NORMALIZATION

A. RELATIONAL MAPPING



B. NORMALIZATION



C. RELATIONAL SCHEMA IN DBDL

```
book(book_id,book_name,TYPE,catagory)
PRIMARY KEY(book_id)
```

```
author(book_id ,author_name)
FOREIGN KEY(book_id) REFERENCES book(book_id)
PRIMARY KEY(book_id,author_name)
```

```
branch(b_id ,NAME,location,phone1,phone2)
PRIMARY KEY(b_id)
```

```
branch_book(book_id ,b_id,availability)
PRIMARY KEY(book_id, b_id)
FOREIGN KEY(book_id) REFERENCES book(book_id)
FOREIGN KEY(b_id) REFERENCES branch(b_id)
```

```
other_services(service_name ,describtion)
PRIMARY KEY(service_name)
```

```
privilage(privilege_id ,book_duration ,loan_period)
PRIMARY KEY(privilege_id)
```

```
sevices_privilage(privilege_id ,service_name)
PRIMARY KEY(privilege_id, service_name)
FOREIGN KEY(privilege_id) REFERENCES
privilage(privilege_id)
FOREIGN KEY(service_name) REFERENCES
other_services(service_name)
```


RELATIONAL SCHEMA IN DBDL CONT.

```
staff(S_id ,F_name ,L_name ,phone,b_id)
PRIMARY KEY(S_id)
FOREIGN KEY(b_id) REFERENCES branch(b_id)
```

```
members(member_id, Password ,Fname,Lname ,member_type
,email,DOB ,privilege_id)
PRIMARY KEY(member_id)
FOREIGN KEY(privilege_id) REFERENCES privilege(privilege_id)
```

```
loan(loan_number,due_date,member_id,b_id)
PRIMARY KEY(loan_number)
FOREIGN KEY(member_id) REFERENCES members(member_id)
FOREIGN KEY(b_id) REFERENCES branch(b_id)
```

```
fine(loan_number,amount ,DATE,state ,discription)
FOREIGN KEY(loan_number) REFERENCES loan(loan_number)
```

```
room(Room_number,size,b_id)
PRIMARY KEY(Room_number)
FOREIGN KEY(b_id) REFERENCES branch(b_id)
```

```
room_book(duration,book_id, DATE,Room_number,member_id)
PRIMARY KEY(Room_number, member_id)
FOREIGN KEY(Room_number) REFERENCES room(Room_number)
FOREIGN KEY(member_id) REFERENCES members(member_id)
```

```
branch_services(service_name,b_id)
PRIMARY KEY(service_name,b_id)
FOREIGN KEY(service_name)REFERENCES other_services(service_name)
FOREIGN KEY(b_id) REFERENCES branch(b_id)
```

4. CREATE DATABASE AND VIEWS

A. DATABASE SCRIPT

```
CREATE TABLE book(  
    book_id INT NOT NULL,  
    book_name VARCHAR(40),  
    TYPE VARCHAR(30),  
    catagory VARCHAR(30),  
    PRIMARY KEY(book_id)  
);  
  
CREATE TABLE author(  
    book_id INT,  
    author_name VARCHAR(30),  
    FOREIGN KEY(book_id)  
REFERENCES book(book_id),  
    PRIMARY KEY(  
        book_id,  
        author_name  
    )  
);  
  
CREATE TABLE branch(  
    b_id INT NOT NULL,  
    NAME VARCHAR(50),  
    location VARCHAR(30),  
    phone1 INT,  
    phone2 INT,  
    PRIMARY KEY(b_id)  
);  
  
CREATE TABLE branch_book(  
    book_id INT NOT NULL,  
    availability VARCHAR(20) NOT NULL,  
    b_id INT NOT NULL,  
    PRIMARY KEY(book_id, b_id),  
    FOREIGN KEY(book_id)  
REFERENCES book(book_id),  
    FOREIGN KEY(b_id) REFERENCES  
branch(b_id)  
);
```

```
CREATE TABLE privilage(  
    privilege_id INT NOT NULL,  
    book_duration VARCHAR(30),  
    loan_period VARCHAR(30),  
    PRIMARY KEY(privilege_id)  
);  
  
CREATE TABLE seviles_privilage(  
    privilege_id INT,  
    service_name VARCHAR(30),  
    PRIMARY KEY(privilege_id,  
        service_name),  
    FOREIGN KEY(privilege_id)  
REFERENCES privilage(privilege_id),  
    FOREIGN KEY(service_name)  
REFERENCES  
other_services(service_name)  
);  
  
CREATE TABLE staff(  
    S_id INT NOT NULL,  
    F_name VARCHAR(30) NOT NULL,  
    L_name VARCHAR(30),  
    phone INT,  
    b_id INT,  
    PRIMARY KEY(S_id),  
    FOREIGN KEY(b_id) REFERENCES  
branch(b_id)  
);  
  
CREATE TABLE other_services(  
    service_name VARCHAR(30) NOT NULL,  
    describtion VARCHAR(40),  
    PRIMARY KEY(service_name) );
```

DATABASE SCRIPT CONT.

```
CREATE TABLE members(  
    member_id INT NOT NULL,  
    Password INT,  
    F_name VARCHAR(30),  
    L_name VARCHAR(30),  
    member_type VARCHAR(30),  
    email VARCHAR(40),  
    DOB DATE,  
    privilege_id INT,  
    PRIMARY KEY(member_id),  
    FOREIGN KEY(privilege_id)  
REFERENCES privileges(privilege_id)  
);
```

```
CREATE TABLE loan(  
    loan_number INT NOT NULL,  
    due_date DATE,  
    member_id INT,  
    b_id INT,  
    PRIMARY KEY(loan_number),  
    FOREIGN KEY(member_id)  
REFERENCES members(member_id),  
    FOREIGN KEY(b_id) REFERENCES  
branch(b_id)  
);
```

```
CREATE TABLE fine(  
    loan_number INT,  
    amount INT,  
    DATE DATE,  
    state VARCHAR(30),  
    discription VARCHAR(40),  
    FOREIGN KEY(loan_number)  
REFERENCES loan(loan_number)  
);
```

```
CREATE TABLE room(  
    Room_number VARCHAR(10) NOT NULL,  
    size VARCHAR(30),  
    b_id INT,  
    PRIMARY KEY(Room_number),  
    FOREIGN KEY(b_id) REFERENCES  
branch(b_id)  
);
```

```
CREATE TABLE room_book(  
    duration VARCHAR(30),  
    book_id INT NOT NULL,  
    DATE DATE,  
    Room_number VARCHAR(10),  
    member_id INT NOT NULL,  
    PRIMARY KEY(Room_number,  
member_id),  
    FOREIGN KEY(Room_number)  
REFERENCES room(Room_number),  
    FOREIGN KEY(member_id)  
REFERENCES members(member_id)  
);
```

```
CREATE TABLE branch_service(  
    service_name VARCHAR(30),  
    b_id INT ,  
    PRIMARY KEY (service_name, b_id),  
    FOREIGN KEY (service_name) REFERENCES  
other_services(service_name),  
    FOREIGN KEY (b_id) REFERENCES  
branch(b_id) );
```

B. VIEW

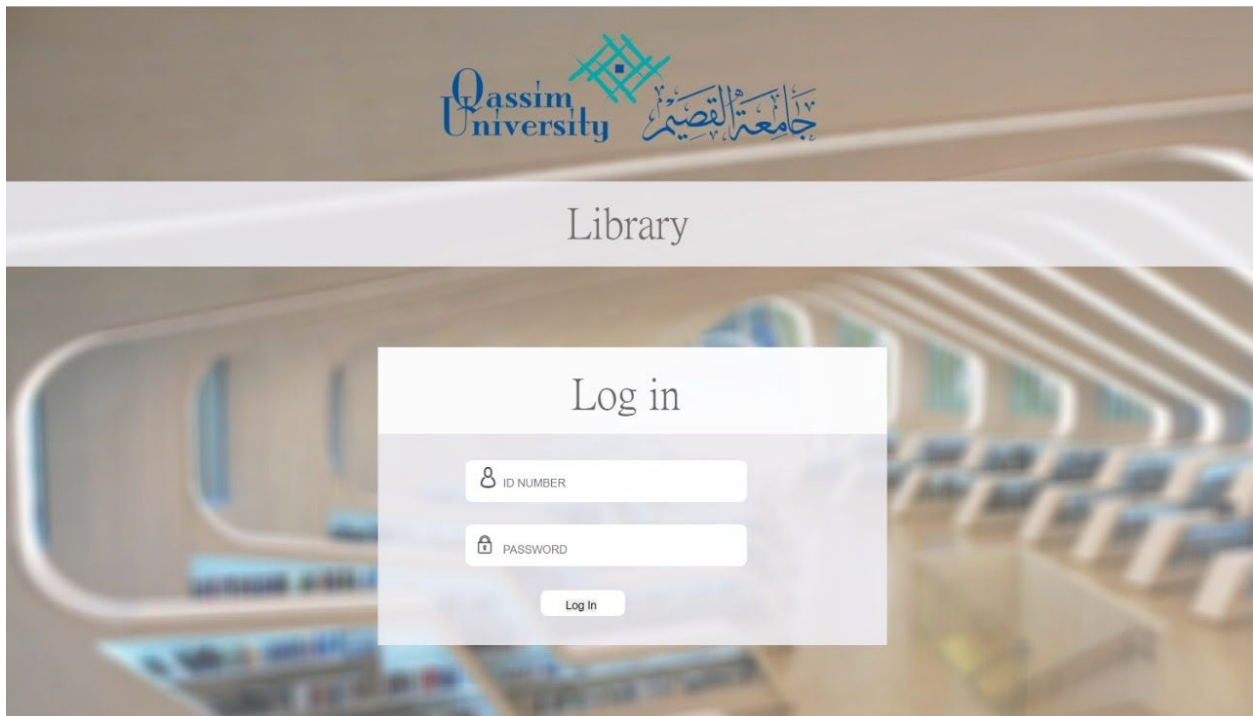
```
CREATE VIEW vBookStatus AS
SELECT
b.book_id,b.book_name,b.type,b.catagory,ba.author_name
FROM book b,branch_book br,author ba
WHERE b.book_id=br.book_id and br.book_id = ba.book_id
and br.availability ='yes';
```

C. DATA ACCESS

```
create roles BookRole
NOT IDENTIFIED ;
grant SELECT ON vBookStatus to BookRole;
```

5. WEB SERVER INTERFACE

THE WEB PAGE



The image shows the login page of the Qassim University Library. At the top, there is a header with the university's logo and name in English and Arabic. Below the header, the word "Library" is centered. The main content area features a "Log in" form with two input fields: "ID NUMBER" and "PASSWORD", each with an icon (a person and a lock respectively). A "Log In" button is positioned below the password field. The background of the page is a blurred image of a library interior with bookshelves.

Qassim University جامعة القصيم

Library

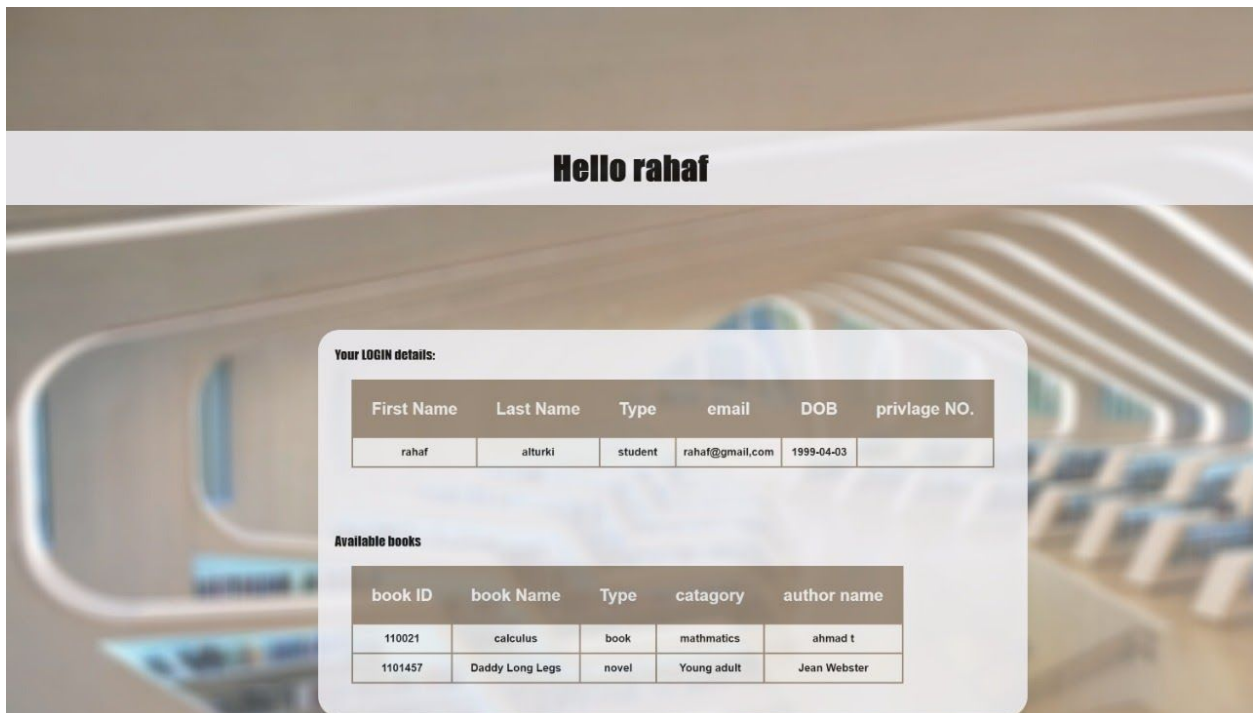
Log in

ID NUMBER

PASSWORD

Log In

After Logging in:



The image shows the user dashboard after logging in. The background is the same blurred library interior. A white box contains the user's login details and a list of available books. The "Your LOGIN details:" section includes a table with columns for First Name, Last Name, Type, email, DOB, and privilege NO. The "Available books" section includes a table with columns for book ID, book Name, Type, category, and author name.

Hello rahaf

Your LOGIN details:

First Name	Last Name	Type	email	DOB	privlage NO.
rahaf	alturki	student	rahaf@gmail.com	1999-04-03	

Available books

book ID	book Name	Type	catagory	author name
110021	calculus	book	mathmatics	ahmad t
1101457	Daddy Long Legs	novel	Young adult	Jean Webster

A.PHP code

```
<body class="background" >
<?php
session_start();

include 'styles.css';
$link =
mysqli_connect("localhost","root","", "library1")
    or die(
mysqli_connect_error("Connection
Failed"));

$username= $_POST["member_id"];
$password= $_POST["Password"];

if (!empty($username)){
    if(!empty($password)){
        $sql= "SELECT * FROM members where
member_id = '$username' ";
        $query = mysqli_query($link, $sql)
        or die( mysqli_connect_error());
        while ($row= mysqli_fetch_array
($query)){
            $DB_username= $row["member_id"];
            $DB_password= $row["Password"];
            if ($username==$DB_username &&
$password== $DB_password){

                echo "<h2> Hello " . $row["F_name"]
. "</h2>";
                echo "<div id='rcorners1'> Your
LOGIN details: ";

                echo "<table
class='cinereousTable'>
                <thead><tr>

                <th> First Name </th>
                <th> Last Name</th>
                <th> Type </th>
                <th> email </th>
                <th> DOB </th>
                <th> privilege NO. </th>
                </tr></thead>";
                $sql= "SELECT * FROM members";
                $query = mysqli_query($link, $sql)
                or die( mysqli_connect_error());
                while ($row= mysqli_fetch_array
($query)){

                    echo "<tbody> <tr> <td> <h4>
".$row["F_name"]. " </h4> </td>";
                    echo "<td> <h4> ".$row["L_name"].
"</h4></td>";
                    echo "<td> <h4>
".$row["member_type"]. " </h4></td>";
                    echo "<td> <h4> ".$row["email"].
"</h4></td>";
                    echo "<td> <h4> ".$row["DOB"].
"</h4></td>";
                    echo "<td> <h4>
".$row["privilege_id"].
"</h4></td></tr></tbody>"; }
                echo "</table >";
                echo "<br><br><br> Available books";
                echo "<table class='cinereousTable'>
                <thead><tr>
                <th> book ID </th>
                <th> book Name</th>
                <th> Type </th>
                <th> catagory </th>
                <th> author name </th>
                </tr></thead>";
                $sql= "SELECT * FROM `vBookStatus`";
                $query = mysqli_query($link, $sql)
                or die( mysqli_connect_error());
                while ($row= mysqli_fetch_array
($query)){
                    echo "<tbody> <tr> <td> <h4>
".$row["book_id"]. " </h4> </td>";
                    echo "<td> <h4> ".$row["book_name"].
"</h4></td>";
                    echo "<td> <h4> ".$row["type"].
"</h4></td>";
                    echo "<td> <h4> ".$row["catagory"].
"</h4></td>";
                    echo "<td> <h4> ".$row["author_name"].
"</h4></td>";
                }
                echo "</table>";
                echo "</div>";
            }
            else{echo "error in login"; }
        }
        else {echo "Enter a password";}
    }
    else {echo "Enter ID number & password";}
    ?>
</body>
```

B. HTML code

```
<!DOCTYPE html>
<html class="background">
<head>
<meta charset="utf-8"/>
<meta http-equiv="X-UA-Compatible"
content="IE=edge"/>
<meta name="viewport"
content="width=device-width,
initial-scale=1.0">
<title>log in </title>

<link rel="stylesheet"
href="styles.css">
</head>
<body >

    <h2 class="library_middle" >
        <span>Library </span>
    </h2>
    
    <svg class="Rectangle_2">
        <rect id="Rectangle_2"
rx="0" ry="0" x="0" y="0" width="610"
height="358">
    </rect>
    </svg>
    <svg class="Rectangle_6">
        <rect id="Rectangle_6"
rx="0" ry="0" x="0" y="0" width="610"
height="104">
    </rect>
    </svg>
    <div id="log_in__p">
        <span>Log in </span>
    </div>

    <form
action="http://localhost//QU/Library.
php " method="post">
        <svg
class="Rectangle_5">
            <rect
```

```
id="Rectangle_5" rx="8" ry="8" x="0"
y="0" width="337" height="50">
        </rect>
    </svg>
    <div id="username_"><input
type="text" class="fadeIn second"
name="member_id" placeholder="ID
NUMBER">    </div>
    <svg class="Rectangle_4">
        <rect id="Rectangle_4"
rx="8" ry="8" x="0" y="0" width="337"
height="50">
    </rect>
    </svg>

    <div id="Password_"><input
type="password" class="fadeIn third"
name="Password"
placeholder="PASSWORD"></div>

    <svg
class="Rectangle_11">
        <rect
id="Rectangle_11" rx="8" ry="8" x="0"
y="0" width="101" height="31"
color="gba(255,255,255,1)">
    </rect>
    </svg>
    <div id="log_in__z">
        <input type="submit"
value="Log In">
    </div>
    </form>
    
    </svg>

    </svg>
    
    </svg>

</body>
</html>
```

C. CSS code

```
<style id="applicationStylesheet"
type="text/css">

    :root {
        --web-view-ids: log_in_;
    }
    * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;
        border: none;
    }

    .background{
background: url('library.jpg');
background-size: 2100px 1176px;
background-repeat: no-repeat;

    }

    #ID1 {
        position: absolute;
        width: 30%;
        left: 35%;
        top: 30px;
        overflow: visible;
    }

    .Rectangle_2 {
        position: absolute;
        overflow: visible;
        width: 610px;
        height: 358px;
        left: 30%;
        top: 409px;
    }

        #Rectangle_2 {
            fill: #f0f0f5;
            opacity: 0.85; }

        #Rectangle_6 {
            fill: rgba(255,255,255,1);
            opacity: .85;
        }
        .Rectangle_6 {
            position: absolute;
            overflow: visible;
            width: 610px;
            height: 104px;
            left: 30%;
            top: 409px;
        }

#log_in_p {
    left: 45.5%;
    top: 435px;
    position: absolute;
    overflow: visible;
    width: 128px;
    white-space: nowrap;
    text-align: left;
    font-family: PMingLiU-ExtB;
    font-style: normal;
    font-weight: normal;
    font-size: 52px;
    color: #666666;
}

#Rectangle_5 {
    fill: rgba(255,255,255,1);
}

.Rectangle_5 {
    position: absolute;
    overflow: visible;
    width: 337px;
    height: 50px;
    left: 37%;
    top: 545px;
}

#username_ {
    left: 40%;
    top: 551px;
    position: absolute;
    overflow: visible;
    width: 105px;
    white-space: nowrap;
    text-align: left;
    font-family: Traditional
        Arabic;

        font-style: normal;
        font-weight: normal;
        font-size: 26px;
        color:
            rgba(168,165,165,0.271);
    }

    #Rectangle_4 {
        fill: rgba(255,255,255,1);
    }
}
```


CSS CODE CONT.

```
.Rectangle_4 {
    position: absolute;
    overflow: visible;
    width: 337px;
    height: 50px;
    left: 37%;
    top: 622px;
}

#Password_ {
    left: 40%;
    top: 631px;
    position: absolute;
    overflow: visible;
    width: 96px;
    white-space: nowrap;
    text-align: left;
    font-family: Traditional
Arabic;
    font-style: normal;
    font-weight: normal;
    font-size: 26px;
    color:
rgba(168,165,165,0.231);
}
#Mask_Group_1 {
    position: absolute;
    width: 24px;
    height: 25px;
    left: 38%;
    top: 553px;
    overflow: visible;
}
#Mask_Group_2 {
    position: absolute;
    width: 16px;
    height: 24px;
    left: 38%;
    top: 634px;
    overflow: visible;
}
#Rectangle_10 {
    fill:
rgba(255,255,255,0.702);
}

#Rectangle_11 {
    fill: rgba(255,255,255,1);
}

.Rectangle_11 {
    position: absolute;
    overflow: visible;
    width: 101px;
    height: 31px;
    left: 43%;
    top: 701px;
}

#log_in__z {
    left: 45%;
    top: 707px;
    position: absolute;
    overflow: visible;
    width: 48px;
    white-space: nowrap;
    text-align: left;
    font-family: PMingLiU-ExtB;
    font-style: normal;
    font-weight: normal;
    font-size: 19px; }

h2{
    text-align: center;
    margin-top: 10%;
    margin-bottom: 5%;
    font-family: Impact, Charcoal,
sans-serif;
    font-size: 40px;

    background: #f0f0f5;
    opacity: 0.85;
    padding: 20px;
}
h2.library_middle{
    margin-top: 14%;
    color: #666666;
    font-family: PMingLiU-ExtB;
    font-style: normal;
    font-weight: normal;
    font-size: 52px;
}
```

CSS CODE CONT.

```
table {  
  
    margin: 20 ;  
    background: #948473;  
    text-align: center;  
}  
table.cinereousTable {  
    background-color: #FFE3C6;  
    width: 100%;  
    text-align: center;  
}  
  
table.cinereousTable thead {  
    padding: 7px 6px;  
    background: #948473;  
    font-size: 20px;  
    font-weight: bold;  
    color: #F0F0F0;  
}  
table.cinereousTable th {  
    padding: 22px 22px;  
font-family: Charcoal, sans-serif;  
}  
table.cinereousTable tbody{  
    background: #F0F0F0; }  
    table.cinereousTable tbody td {  
        font-size: 13px;  
        padding: 9px 8px;  
        font-family: Charcoal, sans-serif;  
    }  
}  
  
#rcorners1 {  
    border-radius: 25px;  
    background: #f0f0f5;  
    opacity: 0.85;  
    padding: 20px;  
    position: fixed;  
    margin-left: 25%;  
    margin-top: 5%;  
    font-family: Impact, Charcoal, sans-serif;  
}  
</style>
```