

|  |  |
| --- | --- |
| **Intake: APD1F2409CS(CYB)** | **Module Code: CT042-3-1-IDB** |

|  |
| --- |
| **Hand Out Date: Tuesday, 22 October 2024 Week 3** |
| **Submission Date:** **Sunday, 22 December 2024 Week 10** |

**TITLE:**

***Community E-Library Management System (CELMS) Part 1***

**Module Lecturer: Mrs. Lai Chew Ping**

**Group No. 2**

|  |  |  |
| --- | --- | --- |
| **No.** | **Name** | **TP** |
| **1** | AHMED MIRAHUSAIN ALVI | TP084807 |
| **2** | MAHRUS SHAMSUL AHSAN | TP085562 |
| **3** | SULTAN ABDULLA OMAR TAKRORI | TP085327 |
| **4** | ABDALLAH MOHAMED MAHMOUD MOHAMED MAHMOUD | TP085097 |
| **5** | MOHAMMED YOUSEF MOHAMMED MOHAMMED | TP085042 |

Table of Contents

**COMMUNITY E-LIBRARY MANAGEMENT SYSTEM (CELMS)………1**

**CHAPTER 1: DATABASE AND DATABASE MANAGEMENT SYSTEM**

**1.1 DISADVANTAGE OF FILE BASED SYSTEM……………………………….2**

**1.2 ADVANTAGE OF FILE BASED SYSTEM …………………………………...4**

**CHAPTER 2: CELMS BUSINESS RULES ..........................................................................5**

**CHAPTER 3: NORMALIZATION**

**3.1 DATA REDUNDANCY ISSUES ……………………………………..................6**

**3.2 UN-NORMALIZED FORM (UNF) ………………………………………..........6**

**3.3 FIRST NORMAL FORM (1NF) …………………………………………...........7**

**3.4 SECOND NORMAL FORM (2NF) ………………………………………..........9**

**3.5 THIRD NORMAL FORM (3NF) …………………………………………….....12**

**CHAPTER 4: ENTITY RELATIONSHIP DIAGRAM (ERD)**

**4.1 ERD………………………………………………………………………….......14**

**4.2 ERD INTERPRETATION……………………………………..........................15**

**CONCLUSION……………………………………………………………………………...16**

**REFERENCES……………………………………………………………………………...17**

**APPENDIX: WORK BREAKDOWN STRUCTURE……………………………...….....19**

**COMMUNITY E-LIBRARY MANAGEMENT SYSTEM (CELMS)**

The Community E-Library Management System (CELMS) is a comprehensive and **user-centric solution** aimed at optimizing the operations of a contemporary e-library. Designed to cater to the diverse requirements of a community centre in Shah Alam, CELMS effectively handles a varied collection of books, magazines, DVDs, and board games, serving an active membership of around 150 people. Embracing a **digital-first strategy**, CELMS improves user experience by providing smooth borrowing procedures, detailed inventory management, and reservation features for meeting rooms.

The system addresses key operational challenges, such as ensuring item availability, enforcing borrowing rules, and calculating overdue fines, all while maintaining an organized and accessible database. CELMS empowers library staff by providing **quick access** to item details, tracking borrowing patterns, and generating insightful reports, fostering a smooth and efficient library experience for both staff and members. Its **scalable design** and adherence to database best practices ensure reliability and adaptability for future needs, making CELMS a cornerstone of community engagement and resource management.

**CHAPTER 1: DATABASE AND DATABASE MANAGEMENT SYSTEM**

***1.1*** *Disadvantages of File Based System*

A file-based system operates by storing and accessing data through individual files, without the use of a centralized database management system (DBMS). This approach may work for **small-scale, standalone applications** but has significant drawbacks when applied to more complex systems like an E-Library Management System. Below, we explore the key limitations of this approach

One major drawback is **data isolation**. In these systems, library operations like inventory, borrowing records, user profiles, and fines are stored in separate, unconnected files. For instance, borrowing data may exist independently from inventory records, complicating tasks like tracking which books are checked out. This lack of integration hinders generating comprehensive reports and prevents advanced features such as analysing user borrowing habits or identifying popular book genres. (Engineering LibreTexts, n.d.)

Additionally, file-based systems suffer from **data redundancy and inconsistency**. The same information, such as book or user details, is often duplicated across multiple files. For example, a book’s title and author may appear in both the inventory and borrowing history files. Changes in one file are not automatically reflected in others, resulting in inconsistencies—one file might indicate a book is available while another shows it is checked out. Such discrepancies force staff to manually update and synchronize files, increasing the risk of errors and inefficiencies. These issues make file-based systems unreliable, reducing their suitability for managing dynamic operations or supporting data-driven decision-making. (GeeksforGeeks, 2023)

To **conclude**, the limitations of file-based systems, including inefficiency, scalability challenges, and security concerns, render them inadequate for managing the complexities of an E-Library Management System. Transitioning to a Database Management System (DBMS) addresses these issues by offering centralized storage for consistency, advanced query capabilities for efficiency, robust security to safeguard sensitive data, and scalability to support future growth. Adopting a DBMS is vital for ensuring reliable, efficient, and secure library management.

***1.2*** *Advantages of Database and DBMS*

A Database Management System (DBMS) provides critical benefits, enhancing the efficiency and reliability of library operations. It ensures **data integrity and consistency** by enforcing rules and constraints to prevent duplication and maintain accurate records. For instance, a DBMS ensures that a book marked as “borrowed” cannot be issued to another member until it is returned. Similarly, updates to member information, such as contact details, are reflected across all related records, standardizing data entry and updates. (GeeksforGeeks, 2024)

The DBMS also excels in **concurrency control**, allowing multiple users to access and update data simultaneously without conflicts or corruption. For example, built-in controls ensure two members cannot borrow the same book at the same time by locking the book record during the transaction. (Iravati Solutions, 2020)

Additionally, a DBMS enhances the **user experience** by offering advanced features like efficient search tools, online reservations, and automated notifications. Members can search for books by various criteria, reserve titles online, and receive timely alerts about due dates or the availability of reserved items. For instance, a member can be notified via email or SMS when a reserved book becomes available for pickup. These features streamline operations and elevate the experience for both library staff and users. (Charbonnet, 2023)

To **conclude**, implementing a DBMS for a Community E-Library Management System (CELMS) significantly enhances operational efficiency, data security, and user experience. By centralizing library operations and providing advanced features such as real-time access, data analytics, and seamless scalability, a DBMS ensures that the library meets the evolving needs of its community. Its advantages benefit all stakeholders, including librarians, members, and administrative staff, while laying a robust groundwork for future growth and technological advancements.

**CHAPTER 2: CELMS BUSINESS RULES**

1. A **Member** can borrow zero, one, or many **Borrowed Items**, but each **Borrowed Item** is borrowed by one and only one member.
2. An **Item** can belong to only one **Category**, but a **Category** can have zero, one, or many **Items.**
3. A **Borrowed Item** is associated with one and only one **Item**, but an **Item** can be borrowed in zero, one, or many borrowing transactions.
4. A **Member** can make zero, one, or many **Reservations**, but each **Reservation** is made by one and only one member.
5. A **Reservation** is associated with one and only one **Room**, but a **Room** can have zero, one, or many **Reservations.**
6. A **Member** can make zero, one, or many **Transactions**, but each **Transaction** is associated with one and only one member. A transaction records the payment date and amount.
7. An **Item** is published by one and only **Publisher**, but a **Publisher** can publish zero, one, or many **Items.**
8. An **Item** can have zero, one, or many **Authors**, and an **Author** can write zero, one, or many **Items.**
9. A **Member** can give zero, one, or many **Feedback** entries, but each **Feedback** entry is given by one and only one member.
10. A **Log** records the activity of one and only one **Member**, but a **Member** can have zero, one, or many **Logs**.
11. A **Popular Genre** can include zero, one, or many **Items**, but an **Item** can belong to one and only one **Popular Genre.**

**CHAPTER 3: NORMALIZATION**

***3.1*** *Data Redundancy Issues*

Data redundancy increases database size and causes issues like:

* **Insertion Anomaly**: Difficulty inserting new rows due to related data being stored together.
* **Deletion Anomaly**: Loss of related data when other data is deleted.
* **Update Anomaly**: Requiring updates across multiple redundant rows.

**Solution**: Normalization minimizes redundancy by organizing data into separate, related entities, reducing problems with data insertion, deletion, and updating.

A screenshot of a computer

Description automatically generated***3.2*** *Un-Normalized Form (UNF)*

**1:**

A screenshot of a computer

Description automatically generated

2:

The table shows a library management system with member details, borrowed books, loan/due dates, overdue days, fines, and total charges, organized for tracking penalties and overdue books.

***3.3*** *First Normal Form (1NF)*

**Transformation from UNF to 1NF:**

To convert from UNF to 1NF we must eliminate any repeating groups. Relational tables must not contain any repeating groups. To do this we must present the data in tabular format where each cell has a single value. Furthermore, eliminate null values by ensuring each repeating group attribute contains an appropriate data value. We must then Identify the primary keys which uniquely identify the attribute value and identify all dependencies

First table: loan details

**Attributes**: MEMBERID, FULL NAME, EMAIL, GENDER, BOOK NO., BOOK TITLE, AUTHOR, GENRE, BORROW DATE, DUE DATE, RETURN DATE, NO. OF DAYS OVERDUE, OVERDUE FEE / PER DAY, TOTAL CHARGE, SUBTOTAL.

In this table all the data is presented in tabular format, and all repeating groups have been eliminated along with all null values. A primary key has also been identified.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| MEMBERID | FULL NAME | E-MAIL | GENDER | BOOK NO. | BOOK TITLE | AUTHOR | GENRE |
| M001 | John Doe | [JD@gmail.com](mailto:JD@gmail.com) | Male | B001 | The Great Gatsby | F. Scott Fitzgerald | Fiction |
| M001 | John Doe | [JD@gmail.com](mailto:JD@gmail.com) | Male | B004 | The Art of War | Sun Tzu | Strategy |
| M001 | John Doe | [JD@gmail.com](mailto:JD@gmail.com) | Male | B002 | National Geographic | Editorial Board | Education |
| M002 | Jane Smith | [JS@gmail.com](mailto:JS@gmail.com) | Female | B002 | National Geographic | Editorial Board | Education |
| M002 | Jane Smith | [JS@gmail.com](mailto:JS@gmail.com) | Female | B006 | Digital Fortress | Dan Brown | Sci-Fi |
| M003 | Ali Khan | [AK@gmail.com](mailto:AK@gmail.com) | Male | B003 | Inception | Christopher Nolan | Sci-Fi |
| M003 | Ali Khan | [AK@gmail.com](mailto:AK@gmail.com) | Male | B006 | Digital Fortress | Dan Brown | Sci-Fi |
| M004 | Sara Lee | [SL@gmail.com](mailto:SL@gmail.com) | Female | B004 | The Art of War | Sun Tzu | Strategy |
| M004 | Sara Lee | [SL@gmail.com](mailto:SL@gmail.com) | Female | B003 | Inception | Christopher Nolan | Sci-Fi |
| M004 | Sara Lee | [SL@gmail.com](mailto:SL@gmail.com) | Female | B005 | Harry Potter | J. K. Rowling | Fiction |

Table 3.2: Data presented in first normalization form

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| BORROW DATE | DUE DATE | RETURN DATE | NO. OF DAYS OVERDUE | OVERDUE FEE/PER DAY | TOTAL CHARGE | SUBTOTALS |
| 8-Nov-24 | 15-Nov-24 | 18-Nov-24 | 3 Days | 2 RM | 6 RM | 21 RM |
| 30-Oct-24 | 3-Nov-24 | 8-Nov-24 | 5 Days | 3 RM | 15 RM | 21 RM |
| 20-Nov-24 | 25-Nov-24 | 25-Nov-24 | 0 Days | 5 RM | 0 RM | 21 RM |
| 6-Nov-24 | 10-Nov-24 | 15-Nov-24 | 5 Days | 5 RM | 25 RM | 37 RM |
| 15-Nov-24 | 20-Nov-24 | 22-Nov-24 | 2 Days | 6 RM | 12 RM | 37 RM |
| 7-Nov-24 | 12-Nov-24 | 17-Nov-24 | 5 Days | 4 RM | 20 RM | 24 RM |
| 1-Nov-24 | 7-Nov-24 | 8-Nov-24 | 1 Day | 4 RM | 4 RM | 24 RM |
| 5-Nov-24 | 14-Nov-24 | 13-Nov-24 | 0 Days | 3 RM | 0 RM | 26 RM |
| 26-Nov-24 | 1-Dec-24 | 5-Dec-24 | 4 Days | 2 RM | 8 RM | 26 RM |
| 9-Nov-24 | 18-Nov-24 | 21-Nov-24 | 3 Days | 6 RM | 18 RM | 26 RM |

Figure 3.2. Data presented in first normalization form (continuation of the first table.)

***3.4*** *Second Normal Form (2NF)*

**Transformation to 2NF:**

Going from 1NF to 2NF requires elimination of the partial dependencies in relation to a primary key. In 1NF, tables violate all the necessary principles of 2NF in cases where some attributes are dependent only on a portion of a composite primary key. Each non-key attribute must be made logically dependent on the entire primary key by restructuring their database into smaller logical tables.

First table: Member Table

**Attributes:** MEMBERID, FULL NAME, EMAIL, GENDER, SUBTOTALS

This table retains all the details of the library members: their full name, email, gender, subtotals and many more information without duplication thereby ensuring that there are no partial dependencies.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMBERID | FULL NAME | E-MAIL | GENDER | SUBTOTALS |
| M001 | John Doe | [JD@gmail.com](mailto:JD@gmail.com) | Male | 21 RM |
| M002 | Jane Smith | [JS@gmail.com](mailto:JS@gmail.com) | Female | 37 RM |
| M003 | Ali Khan | [AK@gmail.com](mailto:AK@gmail.com) | Male | 24 RM |
| M004 | Sara Lee | [SL@gmail.com](mailto:SL@gmail.com) | Female | 26 RM |

Second table: Book Table

**Attributes:** BOOK NO., BOOK TITLE, AUTHOR, GENRE

The **BOOK** table stores information about books available in the library. Each book is uniquely identified by **BOOK NO**, and other attributes such as **BOOK TITLE**, **AUTHOR**, and **GENRE** are directly linked to it

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BOOK NO | BOOK TITLE | AUTHOR | *GENRE* | *OVERDUE FEE/PER DAY* |
| B001 | The Great Gatsby | F. Scott Fitzgerald | Fiction | 2RM |
| B002 | National Geographic | Editorial Board | Education | 5RM |
| B003 | Inception | Christopher Nolan | Sci-Fi | 6RM |
| B004 | The Art of War | Sun Tzu | Strategy | 3RM |
| B006 | Digital Fortress | Dan Brown | Sci-Fi | 6RM |
| B005 | Harry Potter | J. K. Rowling | Fiction | 2RM |

Third Table: Payment Records Table

Attributes: MEMBERID, BOOK NO., BORROW DATE, DUE DATE, RETURN DATE, NO. OF DAYS OVERDUE, TOTAL CHARGE

This table tracks all loan activities and payments. It eliminates partial dependencies by ensuring that each attribute is fully dependent on the composite primary key (MEMBERID and BOOK NO).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| MEMBERID | BOOK NO | BORROW DATE | DUE DATE | RETURN DATE | NO. OF DAYS OVERDUE | TOTAL CHARGE |
| M001 | B001 | 8-Nov-24 | 15-Nov-24 | 18-Nov-24 | 3 Days | 6 RM |
| M001 | B004 | 30-Oct-24 | 3-Nov-24 | 8-Nov-24 | 5 Days | 15 RM |
| M001 | B002 | 20-Nov-24 | 25-Nov-24 | 25-Nov-24 | 0 Days | 0 RM |
| M002 | B002 | 6-Nov-24 | 10-Nov-24 | 15-Nov-24 | 5 Days | 25 RM |
| M002 | B006 | 15-Nov-24 | 20-Nov-24 | 22-Nov-24 | 2 Days | 12 RM |
| M003 | B003 | 7-Nov-24 | 12-Nov-24 | 17-Nov-24 | 5 Days | 20 RM |
| M003 | B006 | 1-Nov-24 | 7-Nov-24 | 8-Nov-24 | 1 Day | 4 RM |
| M003 | B004 | 5-Nov-24 | 14-Nov-24 | 13-Nov-24 | 0 Days | 0 RM |
| M003 | B003 | 26-Nov-24 | 1-Dec-24 | 5-Dec-24 | 4 Days | 8 RM |
| M004 | B005 | 9-Nov-24 | 18-Nov-24 | 21-Nov-24 | 3 Days | 18 RM |

In order to start making the 2NF you need to Check which actions need to be taken, which is to identify all dependencies and to Remove partial dependency.

**Partial Dependency Definition:**

In the context of databases, partial dependency is defined as a circumstance in which non-prime attributes are solely determined by one of the composite primary keys, thus showing a lack of compliance to Second Normal Form. It shows the presence of redundancy. Such or similar types of dependency must be removed to increase the quality of the database design.

|  |  |  |  |
| --- | --- | --- | --- |
| Composite PK | Non-Key Attributes | Check List | Action |
| MEMBERID & BOOKNO. | Full-Name | Is FULL NAME fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, FULL NAME depends on MEMBERID, not BOOK NO. | Partial dependency exists. Remove FULL NAME with determinant (MEMBERID) to form a new table. |
| MEMBERID & BOOKNO. | E-Mail | Is E-MAIL fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, E-MAIL depends on MEMBERID, not BOOKNO. | Partial dependency exists. Remove E-MAIL with determinant (MEMBERID) to form a new table. |
| MEMBERID & BOOKNO. | Gender | Is GENDER fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, GENDER depends on MEMBERID, not BOOK NO. | Partial dependency exists. Remove GENDER with determinant (MEMBERID) to form a new table. |
| MEMBERID & BOOKNO. | SubTotals | Is SUBTOTALS fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, SUBTOTALS depends on MEMBERID, not BOOKNO. | Partial dependency exists. Remove SUBTOTALS with determinant (MEMBERID) to form a new table. |
| MEMBERID & BOOKNO. | BookTitle | Is BOOK\_TITLE fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, BOOK TITLE depends on BOOKNO., not MEMBERID. | Partial dependency exists. Remove BOOK TITLE with determinant (BOOKNO.) to form a new table. |
| MEMBERID & BOOKNO. | Author | Is AUTHOR fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, AUTHOR depends on BOOKNO., not MEMBERID. | Partial dependency exists. Remove AUTHOR with determinant (BOOKNO.) to form a new table. |
| MEMBERID & BOOKNO. | Genre | Is GENRE fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, GENRE depends on BOOKNO., not MEMBERID. | Partial dependency exists. Remove GENRE with determinant (BOOKNO.) to form a new table. |
| MEMBERID & BOOKNO. | OverdueFee/PerDay | Is OVERDUEFEE/PERDAY fully dependent on both (MEMBERID & BOOKNO.)? Answer: No, OVERDUEFEE/PERDAY depends on BOOKNO., not MEMBERID. | Partial dependency exists. Remove OVERDUEFEE/PERDAY with determinant (BOOKNO.) to form a new table. |
| MEMBERID & BOOKNO. | BorrowDate | Is BORROWDATE fully dependent on both (MEMBERID & BOOKNO.)? Answer: Yes, BORROWDATE depends on both (MEMBERID & BOOKNO.) | Full dependency exists, therefore no action needed. BORROWDATE remains in the original table along with its determinant (MEMBERID & BOOKNO.) |
| MEMBERID & BOOKNO. | DueDate | Is DUEDATE fully dependent on both (MEMBERID & BOOKNO.)? Answer: Yes, DUEDATE depends on both (MEMBERID & BOOKNO.) | Full dependency exists, therefore no action needed. DUEDATE remains in the original table along with its determinant (MEMBERID & BOOKNO.) |
| MEMBERID & BOOKNO. | ReturnDate | Is RETURNDATE fully dependent on both (MEMBERID & BOOKNO.)? Answer: Yes, RETURNDATE depends on both (MEMBERID & BOOKNO.) | Full dependency exists, therefore no action needed. RETURNDATE remains in the original table along with its determinant (MEMBERID & BOOKNO.) |
| MEMBERID & BOOKNO. | No.ofDaysOverdue | Is NO.OFDAYSOVERDUE fully dependent on both (MEMBERID & BOOKNO.)? Answer: Yes, NO.OFDAYSOVERDUE depends on both (MEMBERID & BOOKNO.) | Full dependency exists, therefore no action needed. NO.OFDAYSOVERDUE remains in the original table along with its determinant (MEMBERID & BOOKNO.) |
| MEMBERID & BOOKNO. | TotalCharge | Is TOTALCHARGE fully dependent on both (MEMBERID & BOOKNO.)? Answer: Yes, TOTALCHARGE depends on both (MEMBERID & BOOKNO.) | Full dependency exists, therefore no action needed. TOTALCHARGE remains in the original table along with its determinant (MEMBERID & BOOKNO.) |

***3.5*** *Third Normal Form (3NF)*

In 2NF, we handle partial dependencies by ensuring that each non-key attribute is completely dependent on the entire primary key. However, transitional dependencies still exist.

Transitive dependency occurs when a non-key attribute is dependent on another non-key attribute. For example:

* **BOOK NO** **→ GENRE****→ OVERDUE FEE/PER DAY** (OVERDUE FEE/PER DAY is transitively dependent on BOOK NO).
* OVERDUE FEE/PER DAY is a non-key attribute is dependent on GENREthat is another non-key attribute because GENREdetermines OVERDUE FEE/PER DAY.

To address this in 3NF we need to Remove GENRE& OVERDUE FEE/PER DAY to form a new table. However, GENRE needs to remain in the Book table as a reference. This ensures that all attributes are based only on the primary key, achieving 3NF.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMBERID | FULL NAME | E-MAIL | GENDER | SUBTOTALS |
| M001 | John Doe | [JD@gmail.com](mailto:JD@gmail.com) | Male | 21 RM |
| M002 | Jane Smith | [JS@gmail.com](mailto:JS@gmail.com) | Female | 37 RM |
| M003 | Ali Khan | [AK@gmail.com](mailto:AK@gmail.com) | Male | 24 RM |
| M004 | Sara Lee | [SL@gmail.com](mailto:SL@gmail.com) | Female | 26 RM |

**Member**  
 **Attributes: MEMBERID, FULL NAME, EMAIL, GENDER, SUBTOTALS**

This table contains all the information about the library members. Each member's data, including full name, email, and gender, is stored independently without duplication, ensuring that dependency violations do not occur.

|  |  |  |  |
| --- | --- | --- | --- |
| BOOK NO | BOOK TITLE | AUTHOR | *GENRE* |
| B001 | The Great Gatsby | F. Scott Fitzgerald | Fiction |
| B002 | National Geographic | Editorial Board | Education |
| B003 | Inception | Christopher Nolan | Sci-Fi |
| B004 | The Art of War | Sun Tzu | Strategy |
| B006 | Digital Fortress | Dan Brown | Sci-Fi |
| B005 | Harry Potter | J. K. Rowling | Fiction |

**Book**  
 **Attributes: BOOK NO., BOOK TITLE, AUTHOR, GENRE**

The BOOK table stores all the details of the books available in the library. Each book has a unique identifier (book number), and attributes such as book title and author are directly linked to it. The GENRE attribute identifies the book type and corresponds to entries in the fee table.

|  |  |
| --- | --- |
| GENRE | OVERDUE FEE/PER DAY |
| Fiction | 2 RM |
| Strategy | 3 RM |
| Education | 5 RM |
| Sci-Fi | 6 RM |

**Fee**  
 **Attributes: GENRE, OVERDUE FEE/PER DAY**

The FEE table manages overdue charges based on genres. The GENRE attribute ensures that every book genre matches an existing fee category.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| MEMBERID | BOOK NO | BORROW DATE | DUE DATE | RETURN DATE | NO. OF DAYS OVERDUE | TOTAL CHARGE |
| M001 | B001 | 8-Nov-24 | 15-Nov-24 | 18-Nov-24 | 3 Days | 6 RM |
| M001 | B004 | 30-Oct-24 | 3-Nov-24 | 8-Nov-24 | 5 Days | 15 RM |
| M001 | B002 | 20-Nov-24 | 25-Nov-24 | 25-Nov-24 | 0 Days | 0 RM |
| M002 | B002 | 6-Nov-24 | 10-Nov-24 | 15-Nov-24 | 5 Days | 25 RM |
| M002 | B006 | 15-Nov-24 | 20-Nov-24 | 22-Nov-24 | 2 Days | 12 RM |
| M003 | B003 | 7-Nov-24 | 12-Nov-24 | 17-Nov-24 | 5 Days | 20 RM |
| M003 | B006 | 1-Nov-24 | 7-Nov-24 | 8-Nov-24 | 1 Day | 4 RM |
| M003 | B004 | 5-Nov-24 | 14-Nov-24 | 13-Nov-24 | 0 Days | 0 RM |
| M003 | B003 | 26-Nov-24 | 1-Dec-24 | 5-Dec-24 | 4 Days | 8 RM |
| M004 | B005 | 9-Nov-24 | 18-Nov-24 | 21-Nov-24 | 3 Days | 18 RM |

**Payment Records**  
 **Attributes: MEMBERID, BOOK NO., BORROW DATE, DUE DATE, RETURN DATE, NO. OF DAYS OVERDUE, TOTAL CHARGE**

The payment records table tracks all borrow activity and payments. This structure ensures effective management of borrowing records and late fees.

**CHAPTER 4: ENTITY RELATIONSHIP DIAGRAM**

*A diagram of a data flow

Description automatically generatedERD Figure 1:*

A diagram of a computer

Description automatically generated

*ERD Figure 2:*

***ERD Interpretation***

*The Entity-Relationship Diagram of the Community E-Library Management System (CELMS) comprises of* ***12 entities****, each fulfilling a unique function as follows:*

1. **Feedback**: *Gathers information about member’s complaints and suggestion.*
2. **Transactions**: *Maintains record of monetary activities carried out by the members.*
3. **Logs**: *Keeps history of the activities or events accomplished by members in the system.*
4. **Members**: *Indicates the library patrons that checkout items, give suggestions, book items or rooms and carry out payment transactions.*
5. **Reservations:** *Keeps record of rooms set aside by members.*
6. **BorrowedItems:** *Contains the record of what library items members have checked out.*
7. **Rooms:** *Indicates the rooms that can be booked in the library.*
8. **Items:** Maintains comprehensive information about the library's collection, including books, DVDs, magazines, board games and other library resource materials.
9. **Authors:** *Provides insight on the authors of the library items.*
10. **Publishers**: *Provides information on who publishes the library items.*
11. **PopularGenres:** *Shows the types of books together with the number of times they were borrowed.*
12. **Categories:** *Same as the above only that it classifies items into wider dimensions with borrowing conditions.*

**CONCLUSION**

The Community E-Library Management System (CELMS) project is a proper example of the development of a comprehensive system for **storage of information** which is intended to improve the management of libraries. CELMS's use of a relational database eliminates the drawbacks inherent in traditional file systems, making it easier, more reliable, and more expandable.

The project applies the process of identifying business processes and incentivizing the integrity and even minimization of data with structural aggregation, particularly from UNF to 3NF. This task uncovered a well-defined structure consisting of **12 core entities** namely, Members, Feedback, Transactions, Logs, Reservations, Borrowed Items, Rooms, Items, Authors, Publishers, Categories, Popular Genres, and the established links among them. The normalization ensured compliance of CELMS with the best practices of relational databases optimization of requests and strengthening of data consistency.

The Entity Relationship Diagram (ERD) was developed as a graphic representation of the system by illustrating the relationships and constraints in a clear manner. This ERD guarantees that all the rules governing the system of borrowing are each uncompromised, allows for the easy exchange of the items and that the system would be able to handle such operations as reserving items, booking meeting rooms, computing overdue fines, and generating certain types of reports.

However, CELMS neither only tackles issues surrounding operations, but puts customer satisfaction to the next level by delivering features wherever an item is, allowing them to search items efficiently and receive notifications automatically. The designed database can accommodate multiple users simultaneously, preventing CELMS from terminating during periods of high demand.

In conclusion, the CELMS project manages to reach its intended objectives of enhancing the management of the libraries. It provides a sound structure that addresses the problems of librarians and members and still conforms to the extensibility requirement for future development. Having adopted a systematic database and supported advanced features of CELMS demonstrates how libraries can be managed effectively, securely, and oriented to the needs of the users.

# **REFERENCES**

Charbonnet, C. (2023, November 2023). *Reasons Why You Need a Library Management System*. Retrieved from Biblionix: https://www.biblionix.com/reasons-why-you-need-a-library-management-system/

Engineering LibreTexts. (n.d.). *Chapter\_1\_-\_Before\_the\_Advent\_of\_Database\_Systems*. Retrieved from eng.libretexts.org: https://eng.libretexts.org/Courses/Delta\_College/Database\_Design\_-\_NOT\_GOOD/01%3A\_Chapter\_1\_-\_Before\_the\_Advent\_of\_Database\_Systems/1.1%3A\_File-based\_System

GeeksforGeeks. (2023, May 15). *disadvantages-of-file-systems*. Retrieved from www.geeksforgeeks.org: https://www.geeksforgeeks.org/disadvantages-of-file-systems/

GeeksforGeeks. (2024, September 30). *advantages-of-database-management-system*. Retrieved from www.geeksforgeeks.org: www.geeksforgeeks.org/advantages-of-database-management-system/

Iravati Solutions. (2020, May 22). *dbms-advantages-and-disadvantages-limitations-of-file-systems*. Retrieved from www.iravatisolutions.com: https://www.iravatisolutions.com/dbms-advantages-and-disadvantages-limitations-of-file-systems/

**APPENDIX: WORK BREAKDOWN STRUCTURE**

|  |  |  |
| --- | --- | --- |
| **Name (TP)** | **Tasks** | **Signatures** |
| **AHMED MIRAHUSAIN ALVI (TP084807)** | * Database and Database management system * UNF Normalization * Cover page * Entity Relationship Diagram * References * Document Creator | *AHMED* |
| **MOHAMMED YOUSEF MOHAMMED MOHAMMED (TP085042)** | * Database and Database management system * Business Rules * 3NF Normalization and Explanation * Work breakdown structure | ***~~MO~~*** |
| **MAHRUS SHAMSUL AHSAN (TP085562)** | * Database and Database management system * Entity Relationship Diagram * 1NF Normalization and Explanation * Table of contents | ***MSA*** |
| **SULTAN ABDULLA OMAR TAKRORI (TP085327)** | * Database and Database management system * Business Rules * 1NF Normalization and UNF Explanation * CELMS Assignment Introduction | A signature on a black background  Description automatically generated |
| **ABDALLAH MOHAMED MAHMOUD MOHAMED**  **(TP085097)** | * Database and Database management system * Entity Relationship Diagram * 2NF Normalization and Explanation * Conclusion | ***ABDO*** |