**Deliverable 3 – Use Case Modeling**

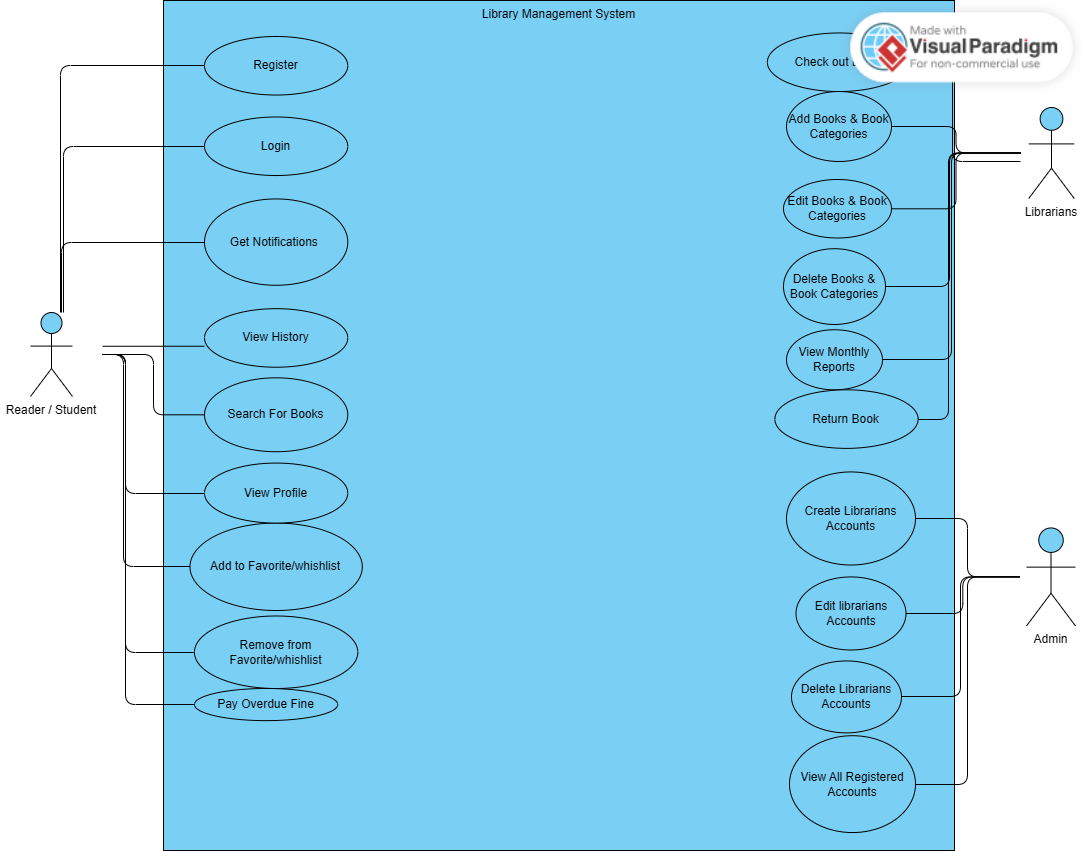
**Team Name:** TechNova

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Team Members  |  |  |  | | --- | --- | --- | | Name | ID | Role | | Ahmed Mohamed Raed | 162023017 | Tester | | Ahmed Salah Ahmed | 162023011 | Backend Developer | | Ibrahim Hashem Mostafa | 162023002 | Project Manager | | Al-Shimaa Ashraf Sayed | 162023041 | Frontend Developer | | Amany Mahmoud Mohamed | 162023043 | Database Designer | |  |  |
|  |  |  |

## Project Title

Library Management System

## Use Case



## 3. Use Case Description

***Section 3: Use Case Descriptions***

*write detailed use case descriptions* ***Use Case 1: Register***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | *Allow a new user to create an account in the system* |
| *Precondition* | *---* |
| *Postcondition* | *User account is created and stored in the database* |
| *Main Flow* | *1. User clicks “Register”  2. Enters required details  3. System validates input  4. System stores user data  5. Registration successful message displayed* |

***Use Case 2: Login***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | *User logs into the system using credentials* |
| *Precondition* | *User is registered* |
| *Postcondition* | *User is granted access to their dashboard* |
| *Main Flow* | *1. User enters email and password  2. System verifies credentials  3. Grants access* |

***Use Case 3: Search For Books***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | Reader/Student, librarians, Admin |
| *Description* | Allow the user to search the entire library catalog to find specific books based on keywords. |
| *Precondition* | The user is logged into the system. |
| *Postcondition* | A list of books matching the search criteria is displayed. |
| *Main Flow* | 1. User navigates to the "Search" page. 2. User enters a search term (e.g., title, author, or category) into the search bar. 3. User clicks the "Search" button. 4. The system queries the database for all books matching the search term. 5. The system displays a results list showing book titles, authors, and availability status. |

***Use Case 4: View Profile***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | This use case allows the Reader to view their personal profile details such as name, email, borrowed books, and account status |
| *Precondition* | |  | | --- | | The Reader must be logged into the system. |  |  | | --- | |  | |
| *Postcondition* | |  | | --- | | The system displays the Reader’s profile information successfully. |  |  | | --- | |  | |
| *Main Flow* | 1. Reader selects the **“View Profile”** option from the menu. 2. The system retrieves the Reader’s personal details from the database. 3. The system displays the Reader’s profile information including name, email, borrowed books, and account history. |

***Use Case 5: Borrowing History***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | |  | | --- | | This use case allows the Reader to view their previous borrow, including item name, date borrowed, and date returned |  |  | | --- | |  | |
| *Precondition* | The Reader must be logged into the system. |
| *Postcondition* | |  | | --- | |  |   *displays the Reader’s borrowing history.* |
| *Main Flow* | 1. Reader selects the ***Borrowing History*** option from the menu. 2. The system retrieves all past borrows from the database. 3. The system displays a list of previous transactions with relevant details (item name, date borrowed, and date returned) |

***Use Case 6: Notification***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | |  | | --- | | This use case allows the Reader to receive and view system notifications such as book availability, due date reminders, or account updates. | |
| *Precondition* | |  |  |  | | --- | --- | --- | | |  | | --- | | The Reader must be logged into the system. |  |  | | --- | |  | | |
| *Postcondition* | |  |  |  | | --- | --- | --- | | |  | | --- | | The system displays the Reader’s notifications successfully. |  |  | | --- | |  | | |
| *Main Flow* | 1. The system generates notifications based on user activity or system events (e.g., a book becomes available, or a due date is near). 2. Reader selects the **“Notifications”** option from the menu or receives a pop-up alert. 3. The system displays a list of recent notifications with relevant details (type, message, and date |

***Use Case 7: Adding to Wish list***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | |  | | --- | |  |   This case allows the Reader to add books to their wishlist for future reference or borrowing. |
| *Precondition* | The Reader must be logged into the system and viewing a detail page. |
| *Postcondition* | |  | | --- | |  |   The selected book is successfully added to the Reader’s Wishlist. |
| *Main Flow* | 1. Reader browses or searches for a book. 2. Reader selects the desired item from the search results or detail page. 3. Reader clicks on the **“Add to Wishlist”** button. 4. The system adds the selected item to the Reader’s wishlist and displays a confirmation message. |

***Use Case 8: Remove From Wishlist***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Reader* |
| *Description* | |  | | --- | |  |   This use case describes how a reader removes a book from their wishlist. |
| *Precondition* | |  |  | | --- | --- | | |  | | --- | | The reader must be logged into the system and have at least one book in their wishlist. | | |
| *Postcondition* | |  |  |  | | --- | --- | --- | | |  | | --- | | The selected book is removed from the reader’s wishlist successfully. |  |  | | --- | |  | |  |  | | --- | |  | |
| *Main Flow* | |  | | --- | | 1. Reader navigates to their wishlist page. 2. Reader selects a book to remove. 3. Reader clicks the “Remove” button. 4. The system removes the book from the wishlist. 5. The system updates and displays the updated wishlis | |

***Use Case 9: Create Librarians Accounts***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Admin* |
| *Description* | This use case describes how a System Administrator creates new librarian accounts to manage library operations. |
| *Precondition* | |  |  |  |  | | --- | --- | --- | --- | | |  |  |  | | --- | --- | --- | | |  | | --- | | The System Administrator is logged into the system with valid credentials. |  |  | | --- | |  | | | |
| *Postcondition* | |  |  | | --- | --- | | |  | | --- | |  | |   A new librarian account is successfully created and stored in the system database. |
| *Main Flow* | 1. The System Administrator navigates to the “Manage Librarians” page. 2. Clicks on the “Add New Librarian” button. 3. Enters the librarian’s information (name, email, username, password, etc.). 4. Clicks “Save” or “Create”. 5. The system validates the information and creates the librarian account. 6. The system displays a confirmation message and updates the list of librarians. |

***Use Case 10: View All Registerd Users***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Admin* |
| *Description* | This use case describes how a System Administrator views a list of all users registered in the system, including readers and librarians. |
| *Precondition* | The System Administrator is logged into the system with valid credentials. |
| *Postcondition* | |  | | --- | |  | | |  | | --- | | The system displays a complete list of all registered users along with their basic information. | | | |
| *Main Flow* | 1. The System Administrator navigates to the “User Management” page. 2. Clicks on “View All Users”. 3. The system retrieves all user records from the database. 4. The system displays the list of users with details such as name, role, and status. 5. The System Administrator can scroll, filter, or search within the list as needed. |

***Use Case 11: Edit Librarians Accounts***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Admin* |
| *Description* | This use case describes how a System Administrator edits the information of an existing librarian account. |
| *Precondition* | The System Administrator is logged into the system and at least one librarian account exists. |
| *Postcondition* | |  | | --- | |  |   The librarian account information is successfully updated in the system database. |
| *Main Flow* | 1. The System Administrator navigates to the “Manage Librarians” page. 2. Selects the librarian account to edit. 3. Clicks on the “Edit” button. 4. Updates the required information such as name, email, or password. 5. Clicks “Save” to confirm changes. 6. The system validates and updates the information in the database. 7. The system displays a confirmation message and the updated librarian details. |

***Use Case 12: Deleting Librarians Accounts***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | *Admin* |
| *Description* | This use case describes how a System Administrator deletes an existing librarian account from the system. |
| *Precondition* | The System Administrator is logged into the system and at least one librarian account exists. |
| *Postcondition* | The selected librarian account is permanently removed from the system database. |
| *Main Flow* | 1. The System Administrator navigates to the “Manage Librarians” page. 2. Selects the librarian account to delete. 3. Clicks on the “Delete” button. 4. The system displays a confirmation message to confirm deletion. 5. The System Administrator confirms the deletion. 6. The system removes the librarian account from the database. 7. The system displays a success message and updates the list of librarians. |

***Use Case 13:*** Add Books and Book Categories

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | Librarian |
| *Description* | |  |  | | --- | --- | | |  | | --- | |  | |   This use case describes how a Librarian adds new books and their corresponding categories to the library system. |
| *Precondition* | |  | | --- | |  |   The Librarian is logged into the system and has permission to manage the book inventory. |
| *Postcondition* | The new book and its category are successfully added to the system database and displayed in the catalog. |
| *Main Flow* | |  |  | | --- | --- | |  | 1. The Librarian navigates to the “Manage Books” section. 2. Clicks on the “Add New Book” button. 3. Enters the book details (title, author, ISBN, publisher, year, etc.). 4. Selects an existing category or clicks “Add New Category” to create a new one. 5. Enters the category name and description (if new). 6. Clicks “Save” to add the book and category. 7. The system validates and stores the data in the database. 8. The system displays a confirmation message and updates the book list. | |

***Use Case 14:*** Edit Books and Book Categories

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | Librarian |
| *Description* | This use case describes how a Librarian edits the details of existing books and their categories in the library system. |
| *Precondition* | The Librarian is logged into the system and at least one book or category exists in the database. |
| *Postcondition* | |  | | --- | |  |   The selected book or category information is successfully updated in the system. |
| *Main Flow* | |  |  | | --- | --- | |  | 1. The Librarian navigates to the “Manage Books” section. 2. Selects the book or category to edit. 3. Clicks on the “Edit” button. 4. Updates the necessary information (such as title, author, category name, or description). 5. Clicks “Save” to confirm changes. 6. The system validates and updates the information in the database. 7. The system displays a confirmation message and the updated record. | |

***Use Case 15:*** Delete Books and Book Categories

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | Librarian |
| *Description* | This use case describes how a Librarian deletes existing books or book categories from the library system. |
| *Precondition* | The Librarian is logged into the system and at least one book or category exists in the database. |
| *Postcondition* | The selected book or category is permanently removed from the system database. |
| *Main Flow* | 1. The Librarian navigates to the “Manage Books” section. 2. Selects the book or category to delete. 3. Clicks on the “Delete” button. 4. The system displays a confirmation message. 5. The Librarian confirms the deletion. 6. The system deletes the selected book or category from the database. 7. The system displays a success message and updates the book or category list. |

***Use Case 16:*** View Monthly Reports

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | Librarian |
| *Description* | This use case describes how a Librarian or System Administrator views the monthly reports that summarize system activities such as issued books, returned books, new registrations, and fines collected |
| *Precondition* | The user (Librarian or System Administrator) is logged into the system with reporting privileges. |
| *Postcondition* | |  | | --- | |  |   The system displays the monthly report for the selected period in a readable format. |
| *Main Flow* | 1. The user navigates to the “Reports” section. 2. Selects “Monthly Reports” from the available options. 3. Chooses the month and year to view. 4. The system retrieves the corresponding report data from the database. 5. The system generates and displays the monthly report including statistics and summaries. 6. The user can download or print the report if needed. |

***Use Case 17: Check Out Book***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | librarians |
| *Description* | This use case describes how a librarian processes a book checkout for a student. |
| *Precondition* | 1. The librarian is logged in. 2. The student is registered and present with their ID.   The book is available. |
| *Postcondition* | 1. The book's status is updated to "Checked Out" in the database. 2. A new entry is created in the Borrowing History table, linking the book to the student and setting a due date.   The student's "Currently Borrowed" list is updated. |
| *Main Flow* | 1. Librarian navigates to the "Check Out" screen. 2. Librarian scans or enters the Student's ID. 3. The system retrieves and displays the student's profile. 4. Librarian scans or enters the book's ID. 5. The system verifies the book is "Available" and that the student is not exceeding their borrow limit. 6. Librarian clicks "Confirm Checkout."   The system updates the database and displays a "Checkout Successful" message. |

***Use Case 18: Return Book***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | librarians |
| *Description* | This use case describes how a librarian processes a book return. |
| *Precondition* | The librarian is logged in.  The book exists in the system and is currently "Checked Out." |
| *Postcondition* | The book's status is updated to "Available."  The BorrowingHistory entry is updated to "Returned."If the book was late, a fine is automatically calculated and added to the student's account. |
| *Main Flow* | 1. Librarian navigates to the "Return Book" screen. 2. Librarian scans or enters the book's ID. 3. The system retrieves the book's record. 4. Librarian clicks "Confirm Return." 5. The system updates the book's status to "Available." 6. The system checks the due date. If it's late, it calculates the fine and associates it with the student who had the book. 7. The system displays a "Return Successful" message. |

***Use Case 19: Pay Overdue Fine***

|  |  |
| --- | --- |
| *Field* | *Description* |
| *Actor* | Reader/Student |
| *Description* | Allows a student to pay their accumulated overdue fines through the online payment module. |
| *Precondition* | The student is logged in.  The student has one or more outstanding fines on their account. |
| *Postcondition* | 1. The student's payment is processed. 2. The corresponding fines are marked as "Paid" in the database. 3. A payment confirmation receipt is generated. |
| *Main Flow* | 1. Student logs in and navigates to the "My Fines" page. 2. The system displays a list of all outstanding fines and a total amount. 3. Student clicks the "Pay Now" button. 4. The system securely redirects the user to the payment form (Online Payments module). 5. Student enters their payment details. 6. The system processes the payment. 7. On success, the system updates the fine status to "Paid" and displays a "Payment Successful" confirmation. |

## 6. References (if any)

Chatgpt: [https://chatgpt.com/share/69034195-5a30-800e-8518-bb008a4a0549](%20https:/chatgpt.com/share/69034195-5a30-800e-8518-bb008a4a0549)

Use Case :<https://online.visual-paradigm.com/>

7. Approval Section (for Instructor use)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |