**Deliverable 2 – Software Requirement Analysis Document**

**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

## 2. Overall Description

**System Perspective:**  
The Library Management System is a desktop application that follows a client-server structure. It allows librarians and students to use the application on their computers while connecting to a central server through a network. The server manages the main database, handles operations such as user registration, book management, borrowing, and returning books, and controls access based on user roles.  
This setup ensures that all library data is centralized, secure, and consistent for every user, while providing an efficient and reliable way to manage daily library activities.

**User Characteristics**:  
The system will be used by librarians and students. Librarians will manage books, users, and reports, requiring basic computer skills. Students will use the system to search, borrow, and return books, needing only basic familiarity with desktop applications.

|  |  |
| --- | --- |
| User Type | Role in System |
| Students, Readers | Will use the system to search for books, check availability, and borrow or return books. |
| librarians | The primary staff user. Will manage book records (add/edit/delete), handle borrowing transactions, and manage student accounts. |
|  |  |
| System Administrator | |  | | --- | |  |  |  | | --- | | Responsible for managing user accounts, maintaining system performance, performing data backups, and ensuring the overall stability and security of the application. | |

## 3. System Modules / Phases

Describe the **main modules or phases** of the project — this helps break the system into manageable components.

|  |  |  |  |
| --- | --- | --- | --- |
| Module Name | Description | Inputs | Outputs / Deliverables |
| User Authentication | Handle login and registration | User credentials | Login success/failure message |
| Dashboard | Display system summary and stats | User Data and permissions | Visual dashboard |
| Reports | Generate performance reports | Data | Downloadable reports |
| Online Payments | |  | | --- | | Manage fine payments for overdue books and other library fees |  |  | | --- | |  | | |  | | --- | | User ID, payment amount, payment method |  |  | | --- | |  | | |  | | --- | | Payment confirmation receipt, updated user balance |  |  | | --- | |  | |
| Manage stock | |  | | --- | | Add, update, delete, and categorize books in the system |  |  | | --- | |  | | |  | | --- | | Book details (title, author, quantity) |  |  | | --- | |  | | |  | | --- | | Updated book inventory and availability status |  |  | | --- | |  | |
| User History | |  | | --- | | Track books borrowed, returned, or renewed by each user |  |  | | --- | |  | | |  | | --- | | User ID, transaction data |  |  | | --- | |  | | |  | | --- | | Borrowing history and transaction logs |  |  | | --- | |  | |
| User Favs (wishlist) | Allow users to save favorite books or mark books to borrow later | |  | | --- | | User ID, selected book IDs |  |  | | --- | |  | | Personalized favorites list or reading list |

(List at least 3–5 modules depending on project size.)

## 4. Functional Requirements

These describe **what the system should do** — its behavior and features.

Each requirement should be **clear, measurable, and testable**.

**Format Example:**

|  |  |  |
| --- | --- | --- |
| Requirement ID | Description | Priority |
| FR1 | The system shall allow users to register using email and password. | High |
| FR2 | The system shall allow Librarians to create, edit, and delete book categories. | High |
| FR3 | The system shall notify users of overdue books and calculate late return fines automatically. | High |
| FR4 | The system shall allow users to search for books by title, author, or category. | High |
| FR5 | The system shall allow System Administrators to create, edit, and delete Librarian accounts. | High |
| FR6 | The system shall allow System Administrators to view all registered users. | Medium |
| FR6 | The system shall generate monthly reports for each Book and Author. | Medium |
| FR8 | The system shall allow users to view their borrowing history and current borrowed books. | Medium |
| FR9 | The system shall allow Librarians to process the return of a book, automatically updating its status to Available. | Medium |
| FR10 | The system shall automatically notify a user within the application when a book in their wishlist becomes available after being returned. | Low |
| FR11 | The system shall allow users to add books to their favorites or wishlist for future reference. | Low |

## 5. Non-Functional Requirements

These describe **how the system should perform** — quality attributes, not specific features.

|  |  |
| --- | --- |
| Category | Description / Requirement |
| Performance | The system should handle up to 100 simultaneous users. |
| Security | Passwords must be encrypted. |
| Usability | A new user shall be able to find and borrow a book within 3 minutes without training |
| Reliability | The system should recover automatically from connection errors. |
| Maintainability | Code should be modular and well-documented. |
| Portability | The system should run on Windows and Linux. |
| Compliance / Legal | The system must comply with all relevant data privacy regarding the storage and usage of patron personal information |

## 6. References (if any)

List any sources, articles, or systems you studied while defining the problem.

7. Approval Section (for Instructor use)

| **Reviewed By** | **Comments** | **And Signature** | **Date** |
| --- | --- | --- | --- |
|  |  |  |  |