**Deliverable 2 – Software Requirement Analysis Document**

**Team Name:** *TechNova*

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

## Project Title

Library Management System

## 2. Overall Description

***System Perspective:*** *The Library Management System is a desktop application designed to automate library operations such as user registration, book management, borrowing and returning, and report generation. It provides role-based access for admins and users, ensuring secure and efficient handling of library data within a standalone environment.*

***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 | who will use the system to search for books, check availability, and borrow or return books. |
| librarians | who will manage book records, handle borrowing and returning transactions, and monitor book availability. |
|  |  |
| System Developers | responsible for maintaining, updating, and ensuring the smooth operation of the system. |
|  |  |

## 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* |
| *1* | *The system shall allow users to register using email and password.* | *High* |
| *2* | *The system shall allow admins to view all registered users.* | *Medium* |
| *3* | *The system shall generate monthly reports for each Book and Author.* | *Medium* |
| *4* | *Create Categories and add books to them* | *High* |
| *5* | |  | | --- | | *The system shall notify users of overdue books and calculate late return fines automatically.* |  |  | | --- | |  | | *High* |
| *6* | |  | | --- | | *The system shall allow users to view their borrowing history and current borrowed books.* |  |  | | --- | |  | | *Medium* |
| *7* | *The system shall allow users to search for books by title, author, or category.* | *High* |
| *8* | |  | | --- | | *The system shall allow users to add books to their favorites or wishlist for future reference.* |  |  | | --- | |  | | *Low* |
| *9* | *The system shall send notifications to user for the books that gets discount or return to stock* | *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* | *The interface should be simple and intuitive.* |
| *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 laws*** *(e.g., GDPR, if applicable) 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** |
| --- | --- | --- | --- |
|  |  |  |  |