Library Borrowing and Reservation System

Hakan Ege Sarıçayır- 220706031

Enes Özdemir-220706057   Yusuf Talha Kamiloğlu- 220706006

1. Project Scenario & Scope

The goal of this project is to design a web-based system that allows library members to search, borrow, and reserve books, while enabling librarians to manage approvals and track book availability. The system focuses on workflow management, validation, and data consistency.

2. Core Actors

- **Library Member:** Searches for books, sends borrowing/reservation requests.
- **Librarian:** Reviews and approves requests, manages book status.
- **System Administrator:** Manages system configurations and users .

3. Functional Requirements

The system must support the following core functions based on the project requirements:

- **FR-01 (Search):** The Member can search for books by **title**, **author**, or **category**.
- **FR-02 (Reserve):** The Member can request to **reserve** a book if it is not currently available.
- **FR-03 (Borrow):** The Member can send a request to **borrow** available books.
- **FR-04 (Validation):** The system must prevent **double borrowing** of the same book copy effectively.
- **FR-05 (Approval):** The Librarian can view, review, and **approve** borrowing and reservation requests.
- **FR-06 (Status Update):** The system must automatically update the **book availability status** upon approval.
- **FR-07 (History):** The Member can view their **current** borrowings and **past** borrowing history.

- **FR-08 (Due Date):** The system must automatically calculate the **return due dates** for borrowed books.

4. Non-Functional Requirements

The system must meet the following quality attributes:

- **NFR-01 (Performance):** The system response time must be **less than 3 seconds**.
- **NFR-02 (Security):** The system must provide **secure login** and authorization mechanisms.
- **NFR-03 (Compatibility):** The application must ensure **Cross-browser compatibility**.
- **NFR-04 (Usability):** The interface must be **user-friendly** and intuitive for all actors.
- **NFR-05 (Reliability): Data integrity and consistency** must be ensured throughout all transactions.

5. JIRA Project Structure (Proposed)

To manage the development process, the project is divided into the following Epics in JIRA:

SPRINT 1: Analysis & Requirements Engineering

(Focus: Defining the project foundation and requirements)

- **Epic 1:** Define Functional Requirements
  - *(Defining the 8 core functions for Members and Librarians)*
- **Epic 2:** Define Non-Functional Requirements
  - *(Defining performance, security, and compatibility criteria)*
- **Epic 3:** Create Requirements Document & GitHub Upload
  - *(Preparing the PDF and uploading it to the /analysis folder)*

SPRINT 2: System Architecture & Modeling

(Focus: Visualizing the system with diagrams)

- **Epic 4:** Design Use Case Diagram

    - *(Modeling actors and system interactions)*
- **Epic 5:** Design Entity Relationship Diagram (ERD)
    - *(Modeling database tables and relationships)*
- **Epic 6:** Design Sequence Diagram
    - *(Modeling the borrowing & reservation workflow)*
- **Epic 7:** Design Activity Diagram (BONUS)
    - *(Modeling the end-to-end process flow for extra credit)*
- **Epic 8:** Upload Diagrams to GitHub
    - *(Exporting and uploading images to the /analysis folder)*

SPRINT 3: Database Implementation

(Focus: Building the technical data infrastructure)

- **Epic 9:** Implement Database Schema (SQL)
    - *(Writing CREATE TABLE scripts based on the ERD)*
- **Epic 10:** Generate Dummy Data
    - *(Creating sample data for testing purposes)*
- **Epic 11:** Upload Database Scripts to GitHub
    - *(Uploading schema.sql to the /database folder)*

SPRINT 4: Frontend UI/UX Design

(Focus: Designing the interface in Figma)

- **Epic 12:** Design Login & Search Screens
    - *(Designing the initial access and book catalog screens)*
- **Epic 13:** Design Borrowing & Reservation Interfaces
    - *(Designing the request forms and book detail pages)*
- **Epic 14:** Upload Figma Links to GitHub
    - *(Saving the project link to /design folder)*