|  |  |
| --- | --- |
| Project name | Library Management System (LMS) |

|  |  |  |
| --- | --- | --- |
| Project author | | |
| № | Names of Participants | FN |
| 1 | Ivan Angelov | 9MI0600190 |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| The **Library Management System (LMS)** offers a solution for managing books, authors, categories, reservations and loans. The LMS allows **Librarians** to manage the library's inventory of books, track loans, and handle user reservations, while **Members** can search for books, review them, reserve them, and check their loan history. The LMS is developed using **Golang, gorilla/mux, gorm,** and **PostgreSQL**. It uses a web-based client powered by frontend framework [**Open UI 5**](https://sdk.openui5.org/topic/ee8726adfdb34d748ed199f0275472f8)to create a dynamic and user-friendly experience.  The system's security and authentication are handled using [**Keycloak**](https://www.keycloak.org/), ensuring robust role-based access control and secure user management. This integration allows for centralized authentication and fine-grained permission control across different user roles. The main roles in the system include:   * **Member** – can view books, authors, and categories, make reservations, create ratings, and track loan history. * **Librarian** – can manage books, authors, categories, reservations, loans and ratings. * **Administrator** – has full control over books, authors, categories, loans, reservations, ratings and users. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| **Register** | Users can register by providing personal information. By default, new users are assigned the "Member" role. | Anonymous User |
| **Manage Users** | Administrators can view and edit user information. | Administrator |
| **View Edit Profile** | Registered user can view and edit their user information. | Registered user |
| **Browse Books** | Users can view books in the library, filter by title, category, author or language, and read reviews. | Registered user |
| **Manage Books** | Librarians and administrators can add, edit, or remove books. | Librarian, Administrator |
| **Browse Authors** | Users can view authors in the library. | Registered user |
| **Manage Authors** | Librarians and administrators can add, edit, or remove authors. | Librarian, Administrator |
| **Browse Categories** | Users can view categories in the library. | Registered user |
| **Manage Categories** | Librarians and administrators can add, edit, or remove categories. | Librarian, Administrator |
| **Loan a Book** | Librarians and administrators can create loans for reserved books. Loans are recorded with start date, due date, and return status. | Librarian, Administrator |
| **Return Book** | Librarians and administrators can return loaned books and update the loan status. | Librarian, Administrator |
| **Reservations** | Users can make reservations for books for a limited time, librarians and administrators can see and extend or delete the reservation. | Registered user |
| **Leave a Review** | Users can review and rate books they have loaned. | Registered user |
| **Track Loan History** | Members can view their loan history, and librarians and administrators can view loan history across users. | Registered user |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| **Register** | Allows users to register providing their credentials and profile information. | [*http://localhost:4000/index.html*](http://localhost:4000/index.html)  *(redirects to keycloak)* |
| **Login** | Allows users to login providing their credentials. | *http://localhost:4000/index.html*  *(redirects to keycloak)* |
| **Home** | Home view with menu with views for books, categories, authors, profile, loans, reservations, ratings for librarians and admins, and users for administrators. | *http://localhost:4000/index.html#/home* |
| **Books** | Displays all books with options to filter and view details. Allows Users to make reservations. Shows full details about a selected book, including reviews and availability. Users can review a book that they have loaned. Librarians and administrators can edit and delete books. | *http://localhost:4000/index.html#/books* |
| **Authors** | Shows all authors with options for edit and delete for librarians and administrators. | *http://localhost:4000/index.html#/authors* |
| **Categories** | Shows all categories with options for edit and delete for librarians and administrators. | *http://localhost:4000/index.html#/categories* |
| **Loans** | Allows Members to view their loans. Shows all loans with options for extending or deleting, and option to return the book for librarians and admins. Loans can be filtered and sorted. | *http://localhost:4000/index.html#/loans* |
| **Reservations** | Allows Members to view and delete their reservations. Shows all reservations with options for extending or deleting, and option to create a loan for librarians and admins. Reservations can be filtered and sorted. | *http://localhost:4000/index.html#/reservations* |
| **Profile** | Allows users to view and edit their personal information. | *http://localhost:4000/index.html#/profile* |
| **Users** | Allows admins to view and edit all users personal information | *http://localhost:4000/index.html#/users* |
| **Ratings** | Allows librarians and admins to view and delete ratings. Rating can be filtered. | *http://localhost:4000/index.html#/ratings* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| **Users** | GET - Administrator (view all registered users)  POST - Administrator | *http://localhost:8080/api/users* |
| **User** | GET - Administrator (any user’s profile), Member (view own profile)  PUT - Administrator (any user’s profile), Member (update own profile, limited fields)  DELETE - Administrator only | *http://localhost:8080/api/users/{userId}* |
| **Books** | GET - Registered user (browse available books with details)  POST - Librarian, Administrator (add new books) | *http://localhost:8080/api/books* |
| **Book** | GET - Registered user (view details of a specific book)  PUT - Librarian, Administrator (update book details)  DELETE - Librarian, Administrator | *http://localhost:8080/api/books/{bookId}* |
| **Categories** | GET - Registered user (view all book categories)  POST - Librarian, Administrator (create new categories) | *http://localhost:8080/api/categories* |
| **Category** | GET - Registered user (view details of a specific category)  PUT - Librarian, Administrator (update category details)  DELETE - Librarian, Administrator | *http://localhost:8080/api/categories/{categoryId}* |
| **Authors** | GET - Registered user (view all authors)  POST - Librarian, Administrator (add new authors) | *http://localhost:8080/api/authors* |
| **Author** | GET - Registered user (view details of a specific author)  PUT - Librarian, Administrator (update author details)  DELETE - Librarian, Administrator | *http://localhost:8080/api/authors/{authorId}* |
| **Loans** | GET - Librarian, Administrator (view all loan records), Member (view own loan history)  POST - Librarian, Administrator (directly record a loan) | *http://localhost:8080/api/loans* |
| **Loan** | GET - Librarian, Administrator (view any loan), Member (view own loan)  PUT - Librarian, Administrator (extend loan, mark as completed)  DELETE - Librarian, Administrator | *http://localhost:8080/api/loans/{loanId}* |
| **Reservations** | GET - Librarian, Administrator (view all reservations), Member (view own reservations)  POST - Registered user (reserve a book) | *http://localhost:8080/api/reservations* |
| **Reservation** | GET - Librarian, Administrator (view any reservation), Member (view own reservations)  PUT - Librarian, Administrator (extend reservation)  DELETE - Member (delete own reservation), Librarian, Administrator (delete any reservation) | *http://localhost:8080/api/reservations/{reservationId}* |
| **Ratings** | GET - Registered user (view all reviews)  POST - Registered user (post a review for a book) | *http://localhost:8080/api/ratings* |
| **Rating** | GET - Registered user (view specific review details)  PUT - Administrator  DELETE - Member (delete own review), Librarian, Administrator (delete any review) | *http://localhost:8080/api/ratings/{ratingId}* |

Libraries used:

 [Logrus](https://github.com/sirupsen/logrus) - Structured logging for Go

 [GORM](https://gorm.io/) - ORM for Go

 [UUID](https://github.com/google/uuid) - UUID generation for Go

 [Gorilla Mux](https://github.com/gorilla/mux) - HTTP request router and dispatcher

 [Viper](https://github.com/spf13/viper) - Configuration management for Go applications

 [CORS](https://github.com/rs/cors) - Middleware for handling CORS in Go

 [GORM PostgreSQL Driver](https://github.com/go-gorm/postgres) - Go driver for PostgreSQL