

System description:

System Domain: Educational Entertainment

The library book borrowing system is a digital platform designed to streamline the management of book borrowing processes within a library. It enables efficient registration and tracking of both books and library members, ensuring the accuracy and completeness of data through various validation checks.

Key features of the system include:

Book registration: Allows librarians to input essential details such as book ID, name, availability, and other relevant information.

Member registration: Enables the registration of library members, capturing details such as name, email, phone number, and age.

Borrowing transactions: Facilitates the borrowing process, allowing members to check out available books and maintaining a record of borrowed items.

Validation checks: Ensures that entered data, including book IDs, member information, and contact details, meet specific criteria to maintain data integrity.

Books Library :

Books Library Model two Modes:

- Book Mode First has: bookID , bookname, publisher, genre, publication year, ISBN, language, description, availabilityBook

Validation:

- **Book ID:** It is a unique identifier used to distinguish each book within the library management system, and it is utilized in searching for and handling books.
Can not be null.
`@NotNull`
Book ID must not be empty.
`@NotEmpty`
Length must be more than 2 characters.
`@Size(min="3" , message=" ID must consist of more than two characters "`
It must be unique for each book.
- **Book Name:** The book name" should match the name displayed on the book cover.
Can not be null.
`@NotNull`
The field must not be empty.
`@NotEmpty`
The name must consist of more than four characters.
`@Size(min=, 5 message=" Name must consist of more than 5 characters"`
It must contain only letters and no numbers.
`@Pattern(regexp=" "[a-zA-Z]+"\ , message = " Book name must contain only letters and no numbers".)`

- Publisher : It is the entity responsible for the production and publication of the book.
 Can not be null.
 @NotNull(message=" Publisher cannot be null")
 The field must not be empty.
 @NotEmpty
 @Pattern(regex="[a-zA-Z]+", message = "Genre must contain only letters and may include spaces")
- Genre: It is a classification used to identify the type or style of the book, such as science fiction, crime, historical, etc
 @NotNull(message = "Genre cannot be null")
 @NotEmpty(message = "Genre must not be empty")
 @Pattern(regex="[a-zA-Z]+", message = "Genre must contain only letters and may include spaces")
- Publication Year: It is the year in which the book was published. The publication year is used to determine the age of the book and its release date.
 @NotNull(message = "Publication year cannot be null")
 @Min(value=0, message="Publication year must be a positive value")
- ISBN: It is an international standard number used to identify books globally and commercially, consisting of 13 digits that help distinguish and track books.
 @NotNull(message = "ISBN cannot be null")
 @NotEmpty(message = "ISBN must not be empty")
 @Pattern(regex="^(97(8|9))?\d{9}(\d|X)\$", message = "ISBN must follow the standard ISBN format")
- Language: It is the language in which the book is written.
 @NotNull(message = "Language cannot be null")
 @NotEmpty(message = "Language must not be empty")
 @Pattern(regex="[a-zA-Z]+", message = "Language must contain only letters and may include spaces")
- Description
 Can not be null
 @NotNull(message = "Description cannot be null")
 The field must not be empty.
 @NotEmpty(message = "Description must not be empty")
- Boolean Availability: is a variable used to represent the availability status of the book, where its value is either true if the book is available, or false if the book is unavailable.
 @NotNull(message = "Availability cannot be null")

Users(reader) Model Second has : name, age, email, phonenumber.

Validation:

- Name
 - Can not be null
`@NotNull(message = "Name cannot be null")`
 - The field must not be empty.
`@NotEmpty(message = "Name must not be empty")`
 - The name must consist of more than four characters.
`@Size(min=, 5 message=" Name must consist of more than 5 characters"`
 - It must contain only letters and no numbers.
`@Pattern(regexp="[a-zA-Z]+", message = " name must contain only letters and no numbers".)`
- Age
 - Can not be null
`@NotNull(message = "Age cannot be null")`
 - The field must not be empty.
`@NotEmpty(message = "Age must not be empty")`
 - Age must be more than 18
`@Min(value=18, message="Age must be more than 18")`
 - Age must be a numeric value
`@Pattern(regexp = "[0-9]+$", message = "Age must be a numeric value")`
- Email:
 - Can not be null
`@NotNull(message = "Email cannot be null")`
 - The field must not be empty.
`@NotEmpty(message = "Email must not be empty")`
 - It must be in a valid email format.
`@Email(message = "Email must be a valid email format")`
- Phone Number:
 - Can not be null.
`@NotNull`
 - The field must not be empty.
`@NotEmpty`
 - It must start with "05".
 - It must consist of exactly 10 digits.
`@Pattern(regexp=" ^05\\d{8}$", message = "Phone number must start with 05 and have 10 digits."`