****

| **Group** **4: Library Management System** | |
| --- | --- |
| **Member :** | Bùi Anh Tuấn  Đoàn Minh Long  Phạm Quang Hưng  Trần Hữu Nguyên |
| **Instructors:** | Tô Hoàng Anh |

**Hà Nội-2022**

# **1. Problem Definition**

## **1.1 Problem Abstraction:**

Using the Internet in the research process has become a common practice by the superiority that it brings such as: faster, more convenient, more efficient and not limited in space and time.

Stemming from those practical needs, our team decided to build an App for the library to create a library management system for the library

**1.2 The Current System:**

Not only the limitation of facilities, but the quality of the library's documents is also a matter of concern. Today, when the requirements for searching documents are higher, requiring depth, expanding into more languages, the library can no longer meet the needs, so people need to switch to be friends with the Internet. With just one click, you can quickly find the information you need. Therefore, many people will choose this way rather than go to the library, then have to make cards and look up books.

Managing books and readers is also more difficult. There is so much literature on books right now that more books will be published and there will be restrictions on taking over that number. About the reader, they will find it harder to find information about the book, even damage or lose the book, making the quality of the book increasingly lower.

## **1.3 The Proposed System:**

**I.Admin**

F1: General display system (total authors, total publishing, total accounts, total readers, total books, total genres).

F2: The system can add books, book categories.

F3: The system can show books(Who add, name, price, co\_year, quantity, description, category’s name, author’s name, publish’s name, status(imported, importing, borrowed, out of stock)), genre, number of books.

F4: The system can edit books(Name, Co\_year, Price, Quantity, Description, Category, Author, Publishing, Status), book genres(Name)

F5: The system can delete books, genres.

F6: The system can add( Name, Sign Name, DOB) author.

F7 The system can show ( Name, Sign Name, DOB, Created at, Updated at) author information.

F8: The system can edit (Name, Sign Name, DOB) author information.

F9: The system can delete the author.

F10: The system can show (Name, Co\_year, Address, Created at, Updated at) publisher information, number of publishers.

F11: The system can add( Name,Co\_year, Address) publisher.

F12: The system can edit (Name, Co\_year, Address) publisher .

F13: The system can delete publisher information.

F14: The system can show (Who borrowed, Account(admin, reader), book, borrow at, refund at, status, time out) borrowed book.

F16: The system can edit(status, time out) borrowed book .

F17: The system can delete borrowed book information .

F18: The system allows the administrator to login to the app.

F19: The system allows the administrator to logout of the app.

F20: The system allows administrators to add accounts (admin and readers).

F21: The system allows editing account information (admin, readers).

F22: The system allows deleting accounts (readers)

.

**II.Librarian**

F23: The system allows librarian login on member account and borrow book in library

**III.Readers**

F24: The system can allow users to register and log out of their accounts.

F25: The system allows users to log in and display the user's dashboard.

F26: The system allows users to choose books to borrow.

F27: The system allows users to edit profile information( Username, Password, fullname, gender, email, birthday, mobile).

F28: The system allows users to search by book name.

F29: The system allows users to display profile information.

## **1.3.1 Boundaries of the System**

- Librarians can only receive notifications of overdue books

- User cannot interact with librarian

## **1.3.2 Hardware and Software Requirements**

Software:

* · Windows 8, 10, or higher
* · JavaFX with Maven
* · FXML
* Scene builder
* NetBeans IDE 14
* JDK 8
* JDBC Driver(MySQL)

Hardware:

* · Intel Core i3 Processor or higher
* · 8GB RAM or higher
* · Color SVGA
* · 8 GB Hard Disk space

## 

# **2. Customer Requirements Specification**

## **2.1 System Requirements**

### **2.1.1 User Of The System:**

* **Librarian**: Can signin reader page and borrow book
* **Readers**: Can view, borrow books
* **Admin**: As the person who has the right to the whole system, performing tasks such as managing customer information, managing books, managing booking

### **2.1.2 System Function:**

* **Admin:**

**F1 General display function:**

* Enter: Enter information about the book to be added
* Processing: Retrieve relevant information from the database.
* Display: display general information..

**F2 Function to add books, book genres:**

* Input: Enter the information about the book, the type of book to add.
* Processing: Add the information just entered to the database.
* Display: Announce the results of the session

**F3 System can show books, genres and book information :**

* Input: Enter information about the book, the genre of the book
* Processing: Retrieve from the database the information related to that book
* Display: Displays relevant details.

**F4 Function to edit book information, book genre**:

* Enter: Re-enter the information about the book, the genre of the book.
* Processing:Check new information, if valid, replace old information, otherwise keep it.
* Display: Announce the results of the session.

**F5 Function to delete books, book genres:**

* Input: The administrator clicks on the "Delete" button next to the title of the book, the type of book to be deleted.
* Processing: Go to the database and delete the books to be deleted. If there is a category, it cannot be deleted.
* Display: Announce the results of the session.

**F6 Function to add author information:**

* Enter: Enter additional information about the author.
* Processing: Add the information just entered to the database.
* Display: Announce the results of the session.

**F7 The system shows author information:**

* Enter: Enter information about the author.
* Processing: Retrieve from the database the information related to that author.
* Display: Displays relevant details.

**F8 Function to edit author information:**

* Enter: Re-enter the author information.
* Handling: Check new information, if valid, replace old information, otherwise keep it.
* Display: Announce the results of the session.

**F9 Function to delete author:**

* Enter: The administrator clicks the “Delete” button next to the author name to be deleted.
* Processing: Retrieve the database and delete the author to be deleted.
* Display: Announce the results of the session.

**F10 Function to show publisher information, number of publishers:**

* Enter: Enter the publisher information.
* Processing: Retrieve from the database the information related to that publisher.
* Display: Displays relevant details.

**F11 Function to add publisher information:**

* Enter: Enter additional information about the publisher.
* Processing: Add the information just entered to the database.
* Display: Announce the results of the session.

**F12 Function to edit publisher information:**

* Enter: Re-enter the publisher information.
* Handling: Check new information, if valid, replace old information, otherwise keep it.
* Display: Announce the results of the session.

**F13 Function to delete publisher:**

* Input: The administrator clicks on the "Delete" button of the publisher to be deleted.
* Handling: Retrieve the database and delete the publisher to be deleted.
* Display: Announce the results of the session.

**F14 Function to show borrow information:**

* Enter: Enter the borrowed book information.
* Processing: Retrieve from the database the information related to that borrowed book.
* Display: Announce the results of the session.

**F15 Function to add borrow information:**

* Enter: Enter additional information about the borrowed book.
* Processing: Add the information just entered to the database.
* Display: Announce the results of the session.

**F16 Function to edit borrow information:**

* Enter: Re-enter the borrowed book information.
* Handling: Check new information, if valid, replace old information, otherwise keep it.
* Display: Announce the results of the session

**F17 Function to delete borrow information:**

* Input: The administrator clicks on the “Delete” button of the borrowed book to be deleted.
* Handling: Retrieve the database and delete borrowed book.
* Display: Announce the results of the session.

**F18 Function for admin login:**

* Enter: Username and password.
* Process: Access the database and compare the name and password with the stored information.
* Display: Admin dashboard

**F19 Function for admin logout:**

* Enter: Click the sign out button.
* Action: Delete logged in username and password.
* Display: Login page.

**F20 Functions for administrators to add admin or reader accounts:**

* Enter: The information of the account to be added.
* Processing: Add the information just entered to the database.
* Display: Notification of work results.

**F21 Functions for administrators to edit admin or reader accounts:**

* Enter: Re-enter the information of the account to be edited.
* Handling: Check new information, if valid, replace old information, otherwise keep it.
* Display: Notification of work results.

**F22 Functions for administrators to delete reader accounts:**

* Enter: Press the "Delete account" button of the account to be deleted.
* Handling: Delete account information in the database.
* Display: Notification of work results.
* **Librarian:**

**F23 Function for librarian to login on member account and borrow book in library:**

* Enter: Name and password.
* Process: Access the database and compare the name and password with the stored information.
* Display: Displays the admin dashboard .

* **Reader:**

**F24 Function to allow users to log out and register:**

* Enter: (Log Out) Click the “Sign Out” button,

(Register) customer information.

* Action: (Logout) Delete logged in username and password,

(Register) Add user's information to the database

* Display: Display the session result message.

**F25 The function that allows the user to login and display the user's dashboard:**

* Enter: Enter your name and password,
* Process: Retrieve the database and compare the user's name and password with the stored information
* Display: Displays the user’s dashboard.

**F26 The function allows users to choose books:**

* Enter: Select the information about the book,
* Processing: Retrieve into the database to retrieve the books with selected information
* Display: Displays the results.

**F27 The function allows users to edit profile information:**

* Enter: Re-enter the borrowed profile information,
* Processing: Check new information, if valid, replace old information, otherwise keep it
* Display: Display the session result message.

**F28 The function allows users to search by book name:**

* Description: Search for information about books by name
* Enter: Ask guests to enter the name of the book they are looking for.
* Processing: Accessing the database, filtering out the books with attributes that match what the customer entered.
* Display: Displays information about the book if it is appropriate, or gives a book not found error if there are no books that match what the customer entered

.

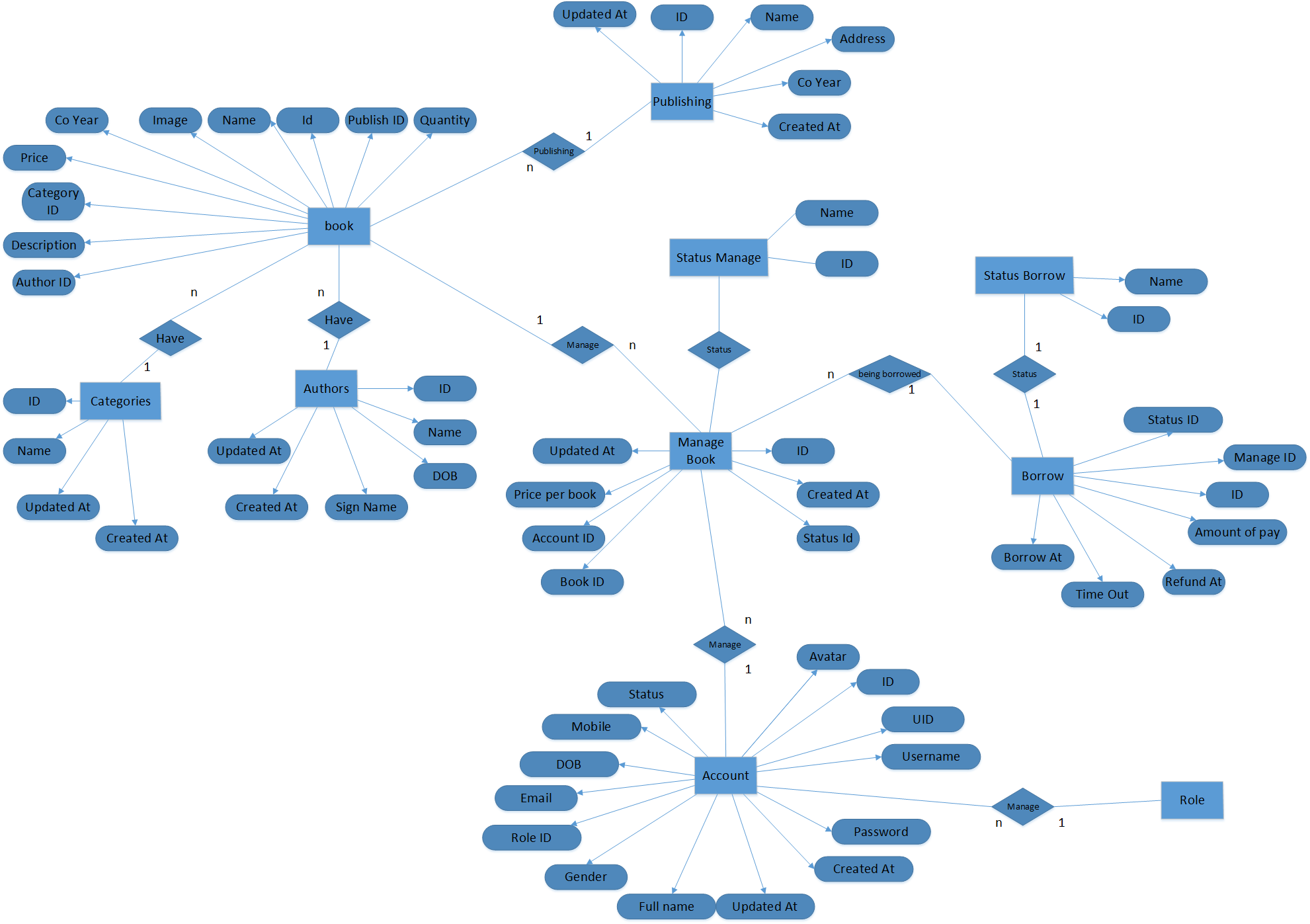
**F29 Function to display profile information:**

* Enter: No need to enter .
* Processing: Get from the database of reader information.
* Display: Notification of expiration time if it is exceeded**.**

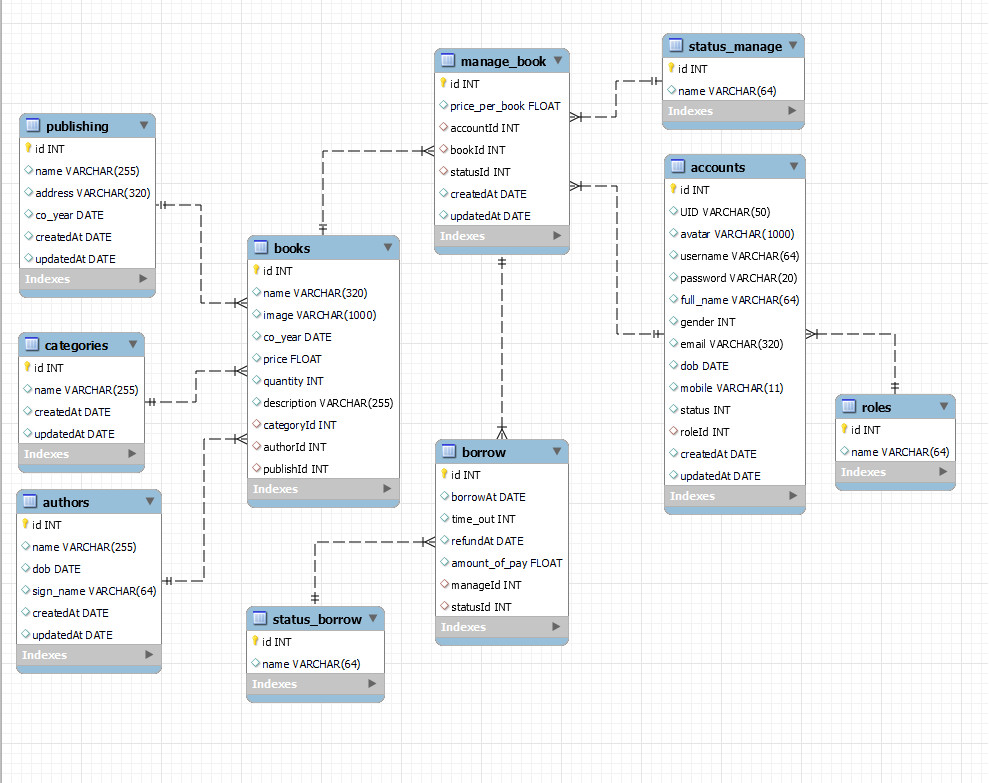
.

## **2.2 System Design:**

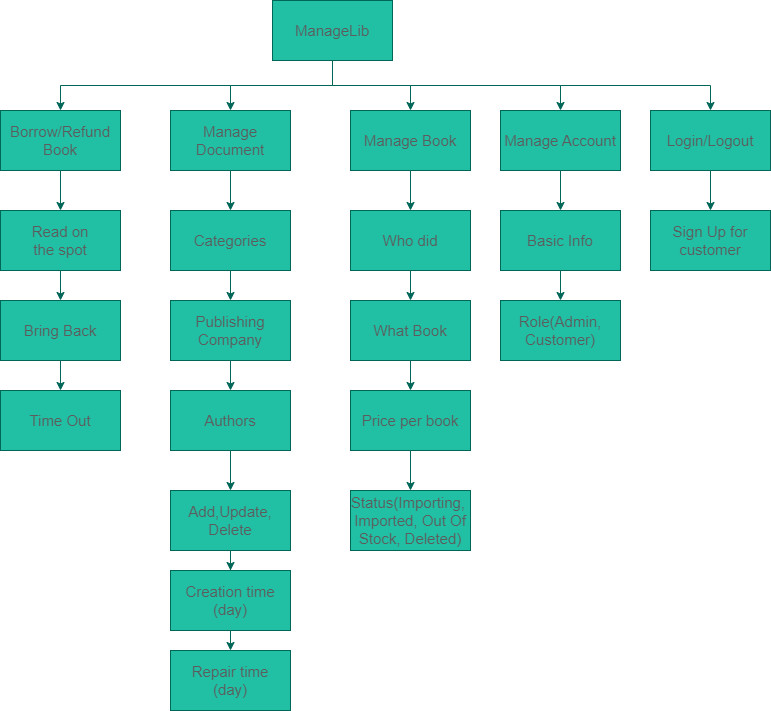
**2.2.1 Entity Relationship Diagram:**

****

**2.2.2 Database design:**

****

**2.2.3 Site Map:**

****

**2.2.3 Table detail**

**-Publishing:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 30 |  |
| address | VARCHAR | 255 |  |
| co\_year | DATE |  | Not Null |
| created\_at | DATE |  | Not Null |
| updated\_at | DATE |  | Not Null |

**-Categories:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 255 | Not Null |
| created\_at | DATE |  | Not Null |
| updated\_at | DATE |  | Not Null |

**-Authors:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 255 | Not Null |
| DOB | DATE |  |  |
| sign\_name | VARCHAR | 64 | Not Null |
| created\_at | DATE |  | Not Null |
| updated\_at | DATE |  | Not Null |

**-Books:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 320 | Not Null |
| image | VARCHAR | 1000 |  |
| co\_year | DATE |  | Not Null |
| price | FLOAT |  | Not Null |
| quantity | iNT |  | Not Null |
| description | VARCHAR | 255 | Not Null |
| category\_id | INT |  | Not Null |
| author\_id | INT |  | Not Null |
| publish\_id | INT |  | Not Null |

**-Status borrow:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 64 | Not Null |

**-Manage\_book :**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| price\_per\_book | FLOAT |  | Not Null |
| account\_id | INT |  | Not Null |
| book\_id | INT |  | Not Null |
| status\_id | FLOAT |  | Not Null |
| created\_at | DATE |  | Not Null |
| updated\_at | DATE |  | Not Null |

**-Borrow:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| borrow\_at | DATE |  | Not Null |
| time\_out | INT |  | Not Null |
| refund\_at | DATE |  | Not Null |
| amount\_of\_ pay | FLOAT |  |  |
| manager\_id | INT |  | Not Null |
| status\_id | INT |  | Not Null |

**-Status\_manager:**

| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 64 |  |

**-Role:**

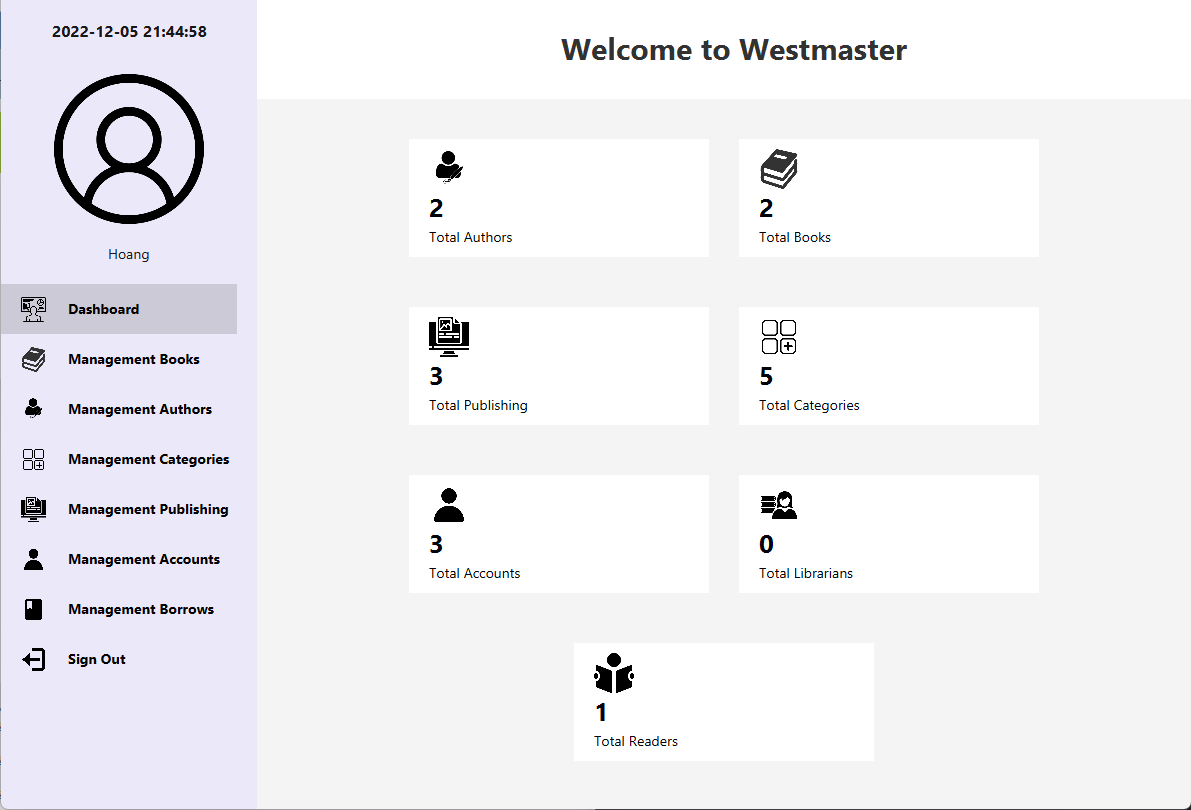
| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| id | INT |  | Primary Key |
| name | VARCHAR | 320 |  |

**-Accounts:**

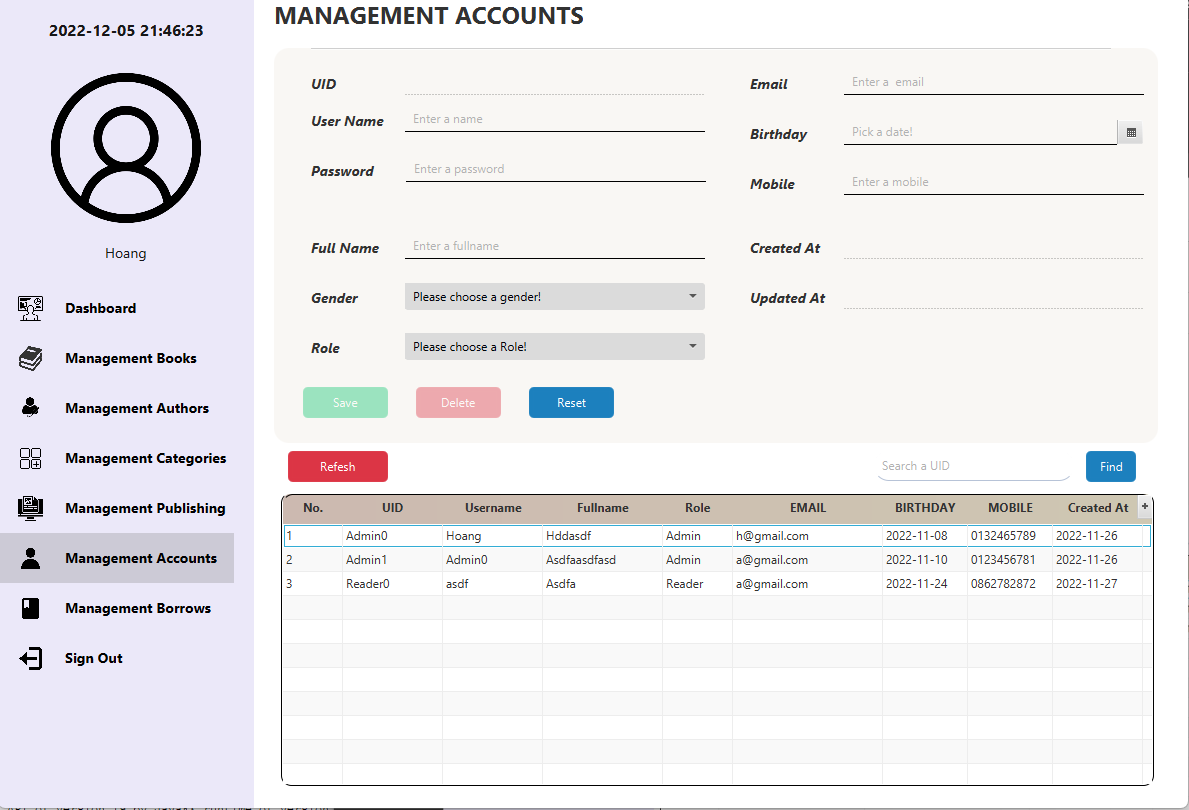
| **Properties** | **Data type** | **Large** | **Core** |
| --- | --- | --- | --- |
| Id | INT |  | Primary Key |
| UID | INT |  | Not Null |
| avatar | VARCHAR | 1000 |  |
| username | VARCHAR | 64 |  |
| password | VARCHAR | 20 |  |
| full\_name | VARCHAR | 64 |  |
| gender | INT |  |  |
| email | VARCHAR | 320 |  |
| dob | DATE |  |  |
| mobile | VARCHAR | 11 |  |
| status | INT |  |  |
| role\_id | INT |  |  |
| created\_at | DATE |  |  |
| updated\_at | DATE |  |  |

**3. Interface**

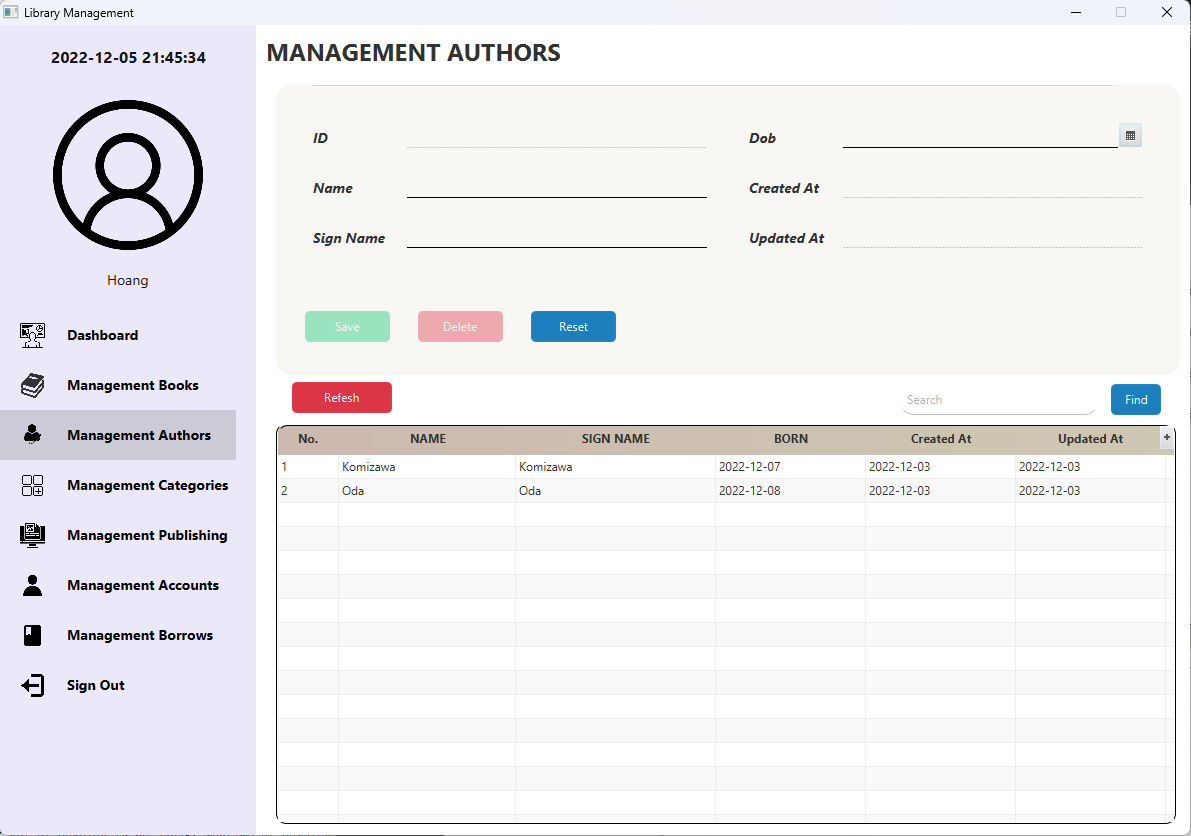
**Admin Dashboard**

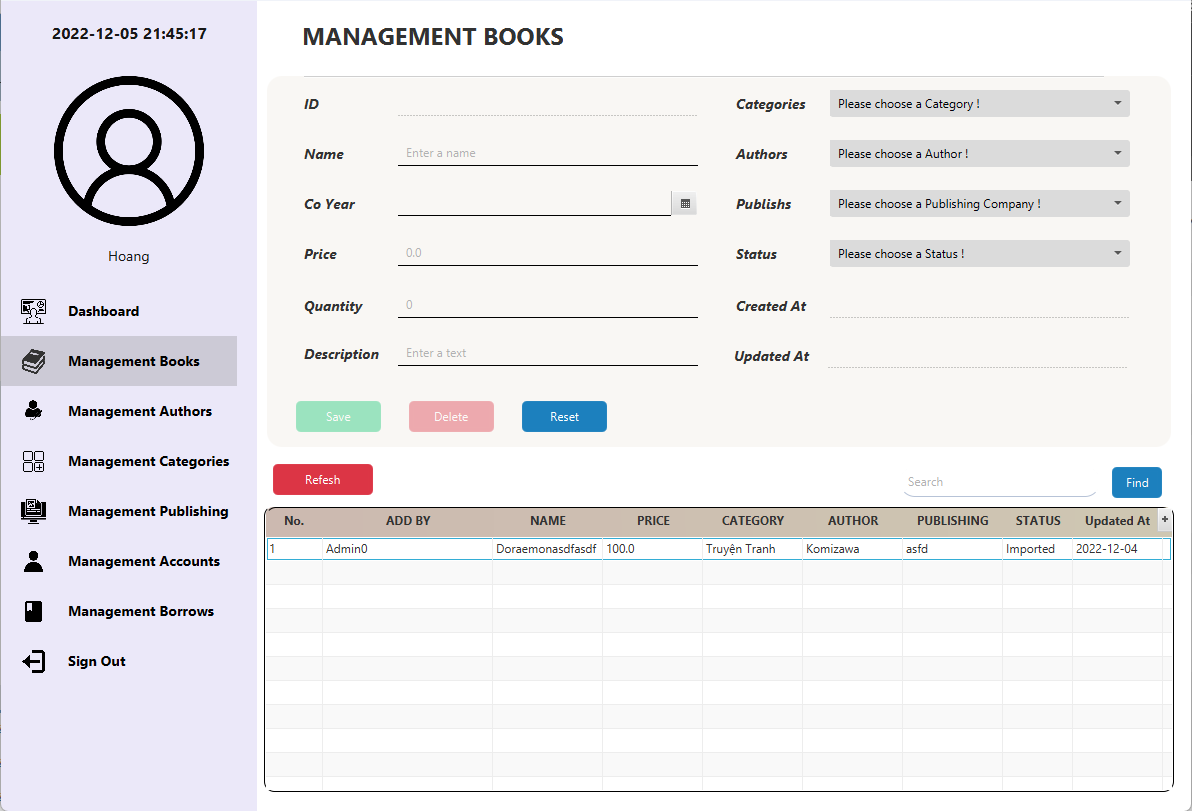


**Admin Manager Account**

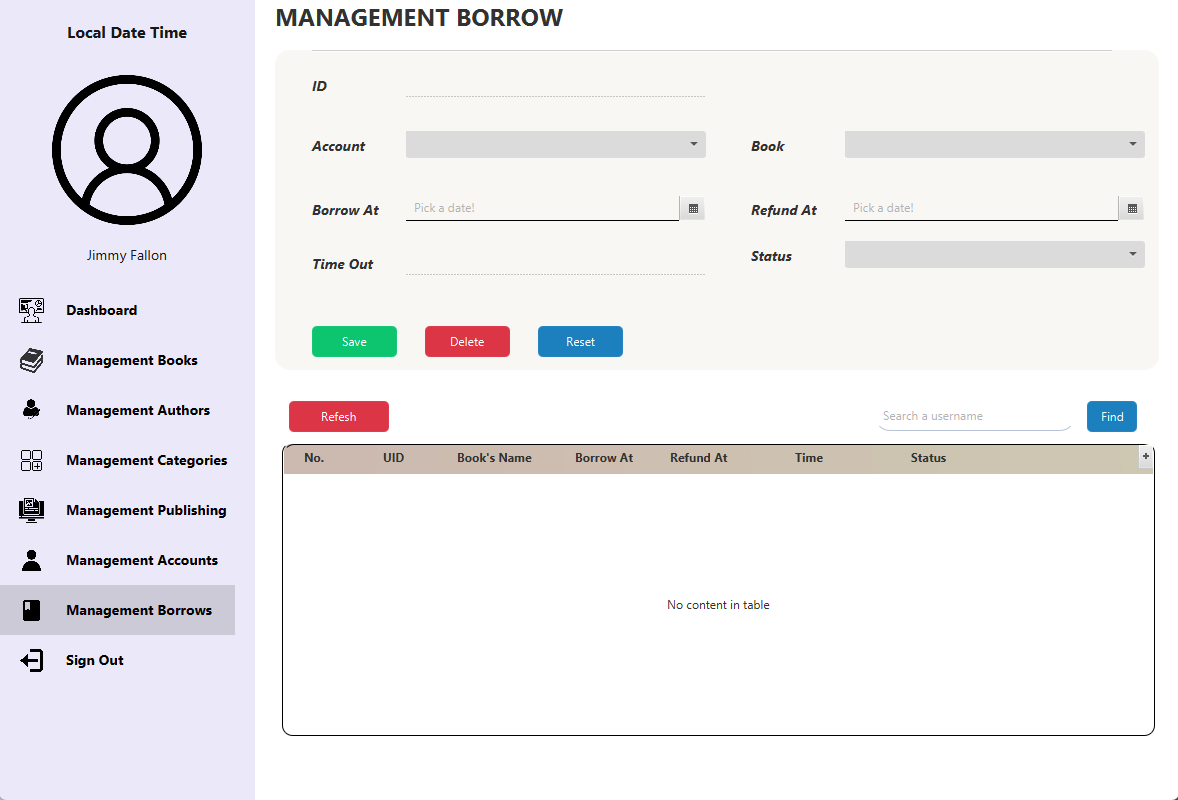


**Admin Manage Authors**

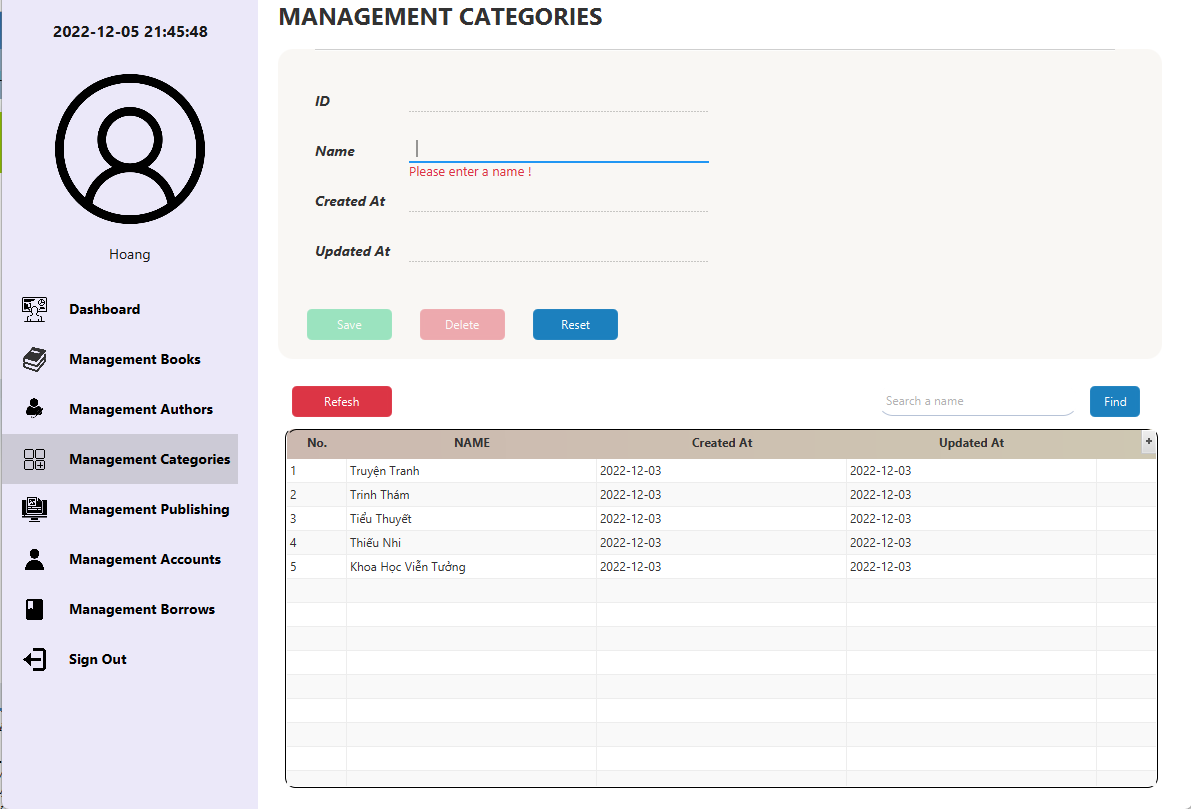
**Admin Manage Book**



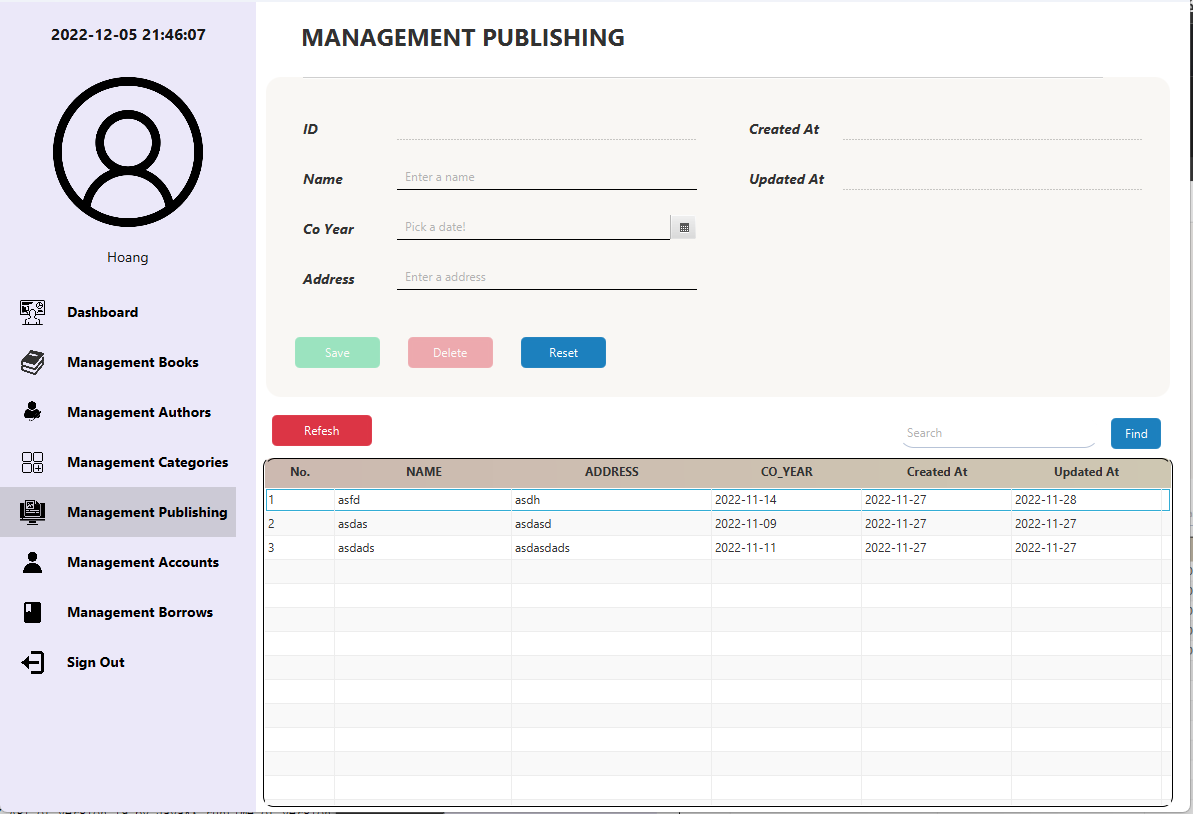
**Admin Manage Borrows**



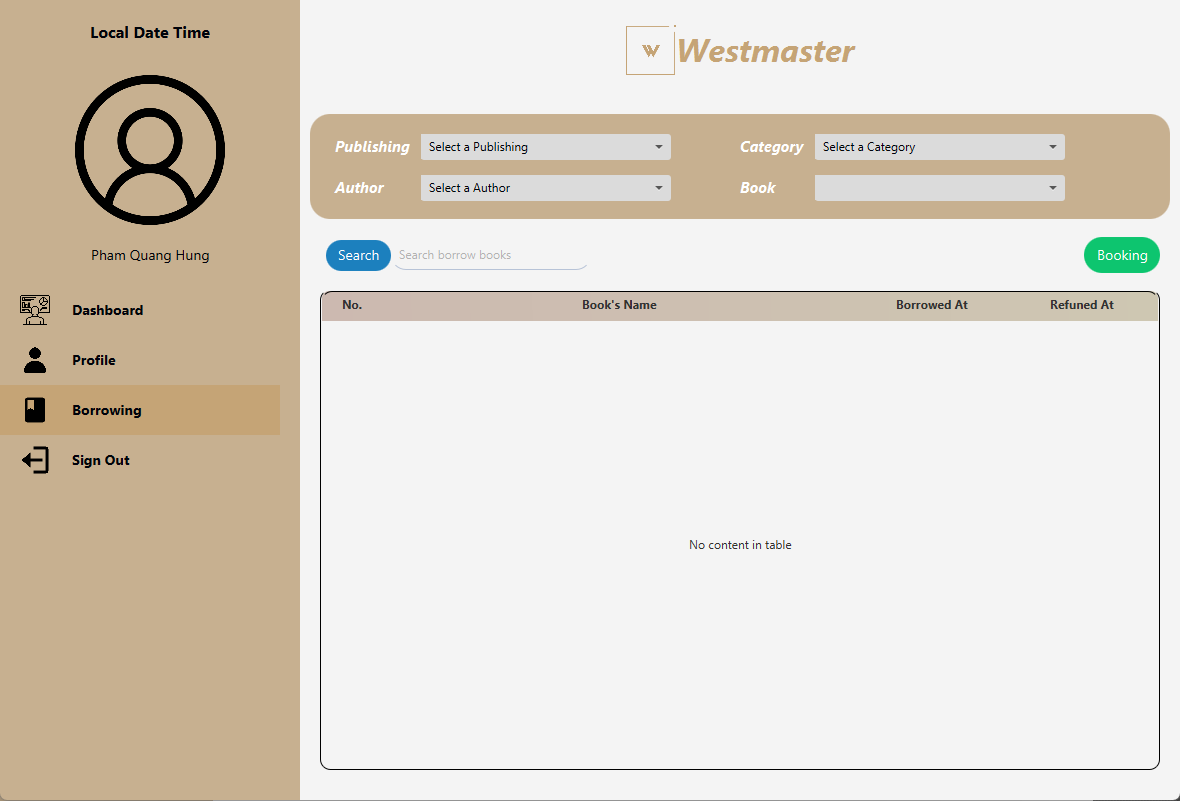
**Admin Manage Categories**



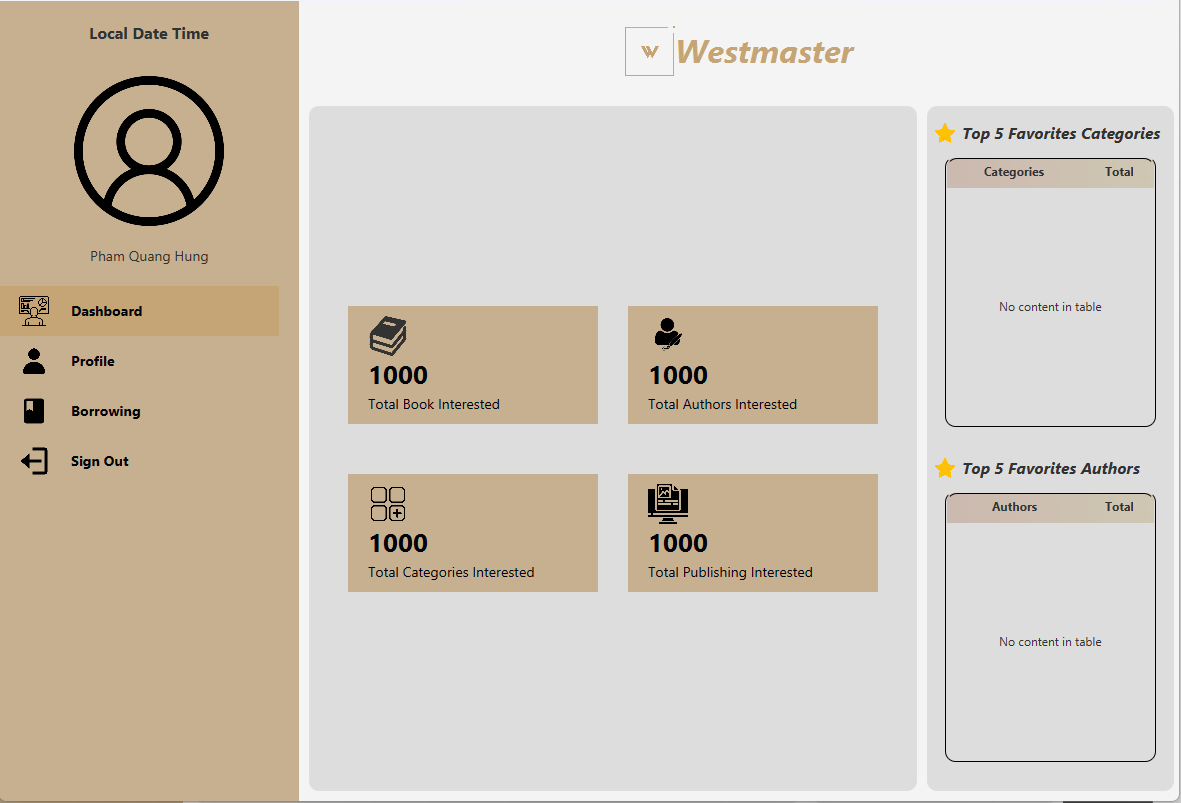
**Admin Manage Publishing**



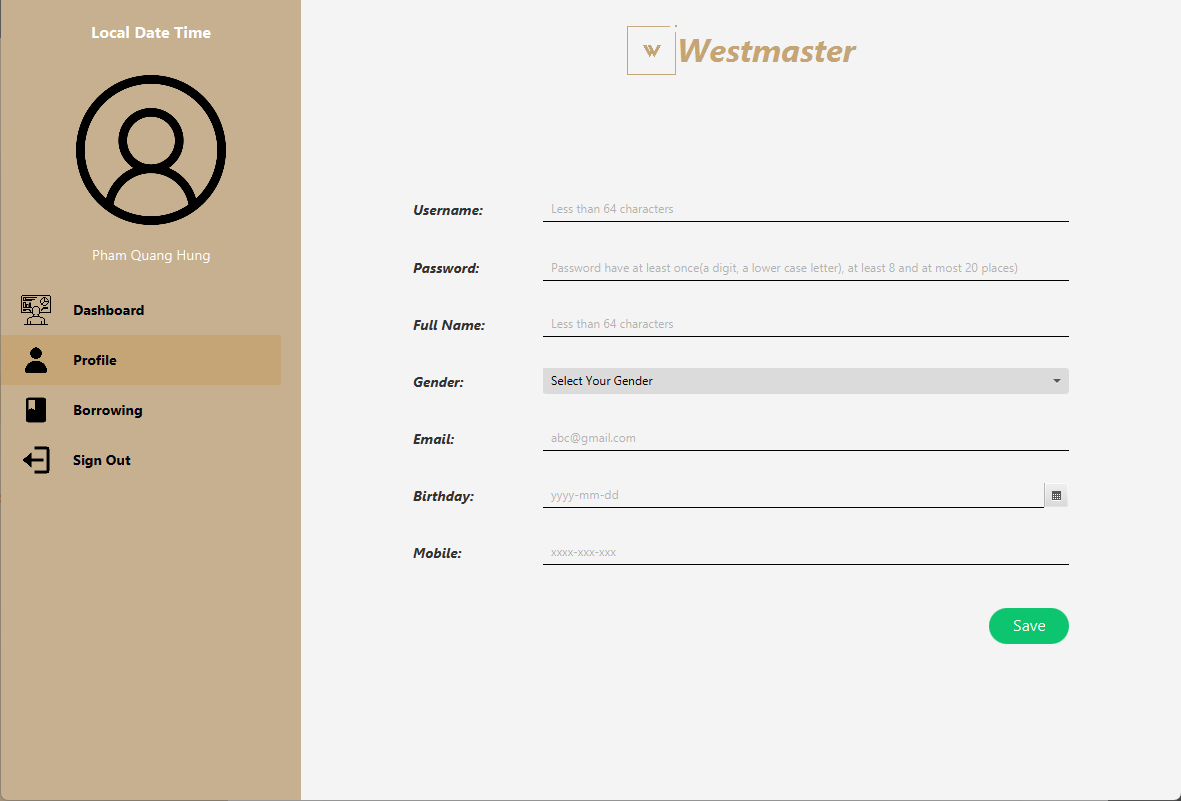
**Customer Borrowing**

****

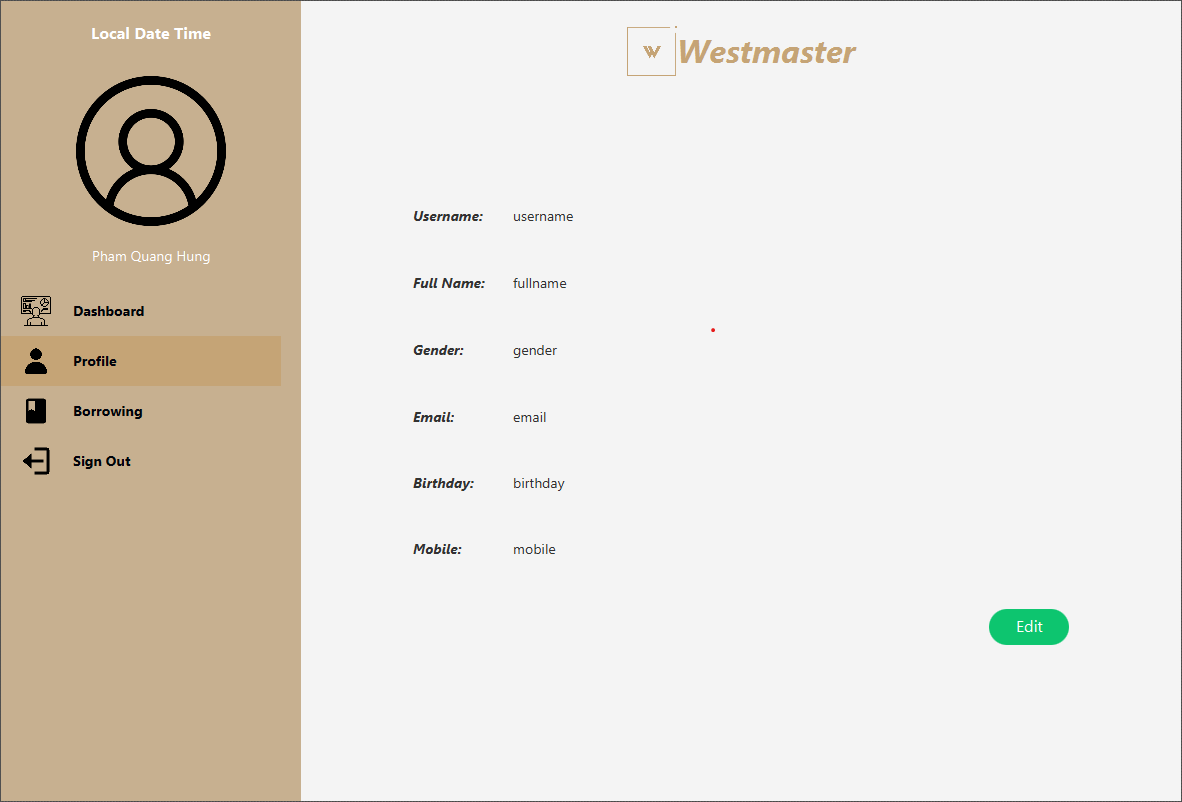
**Customer Dashboard**

****

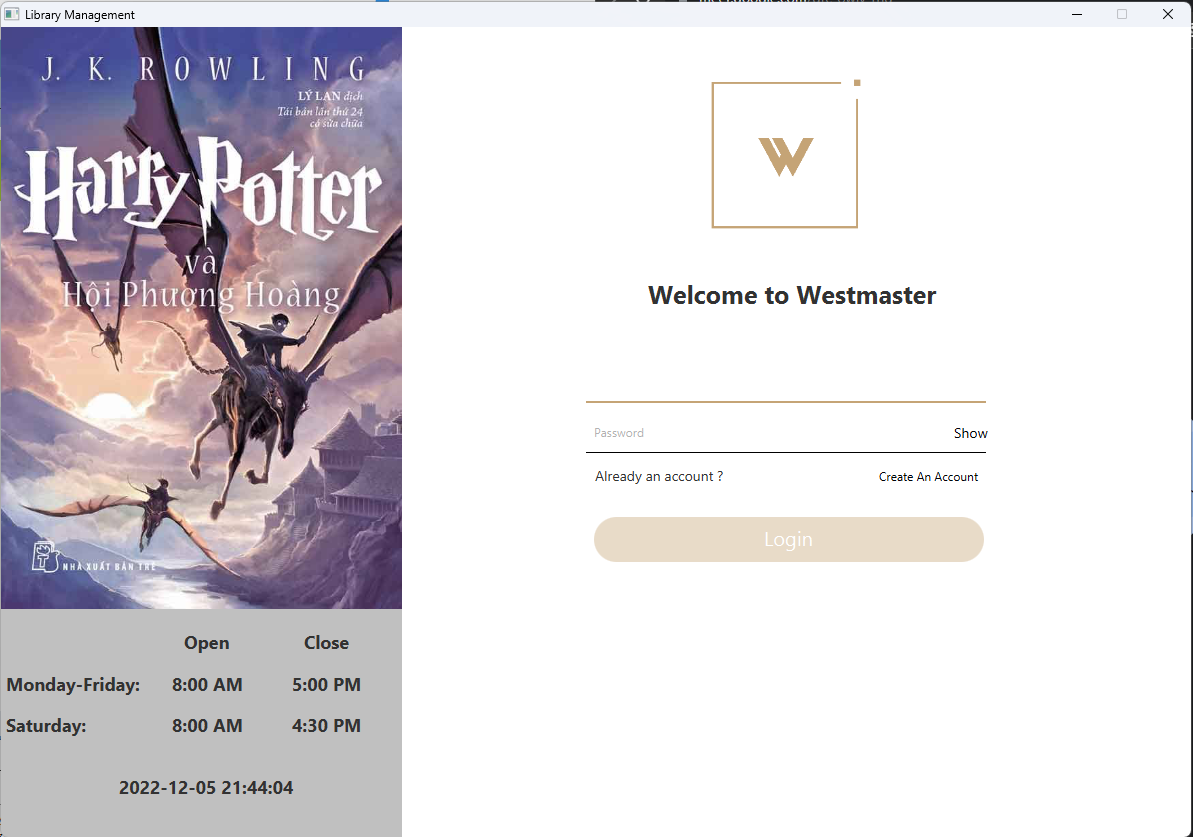
**Customer Edit Information Profile**

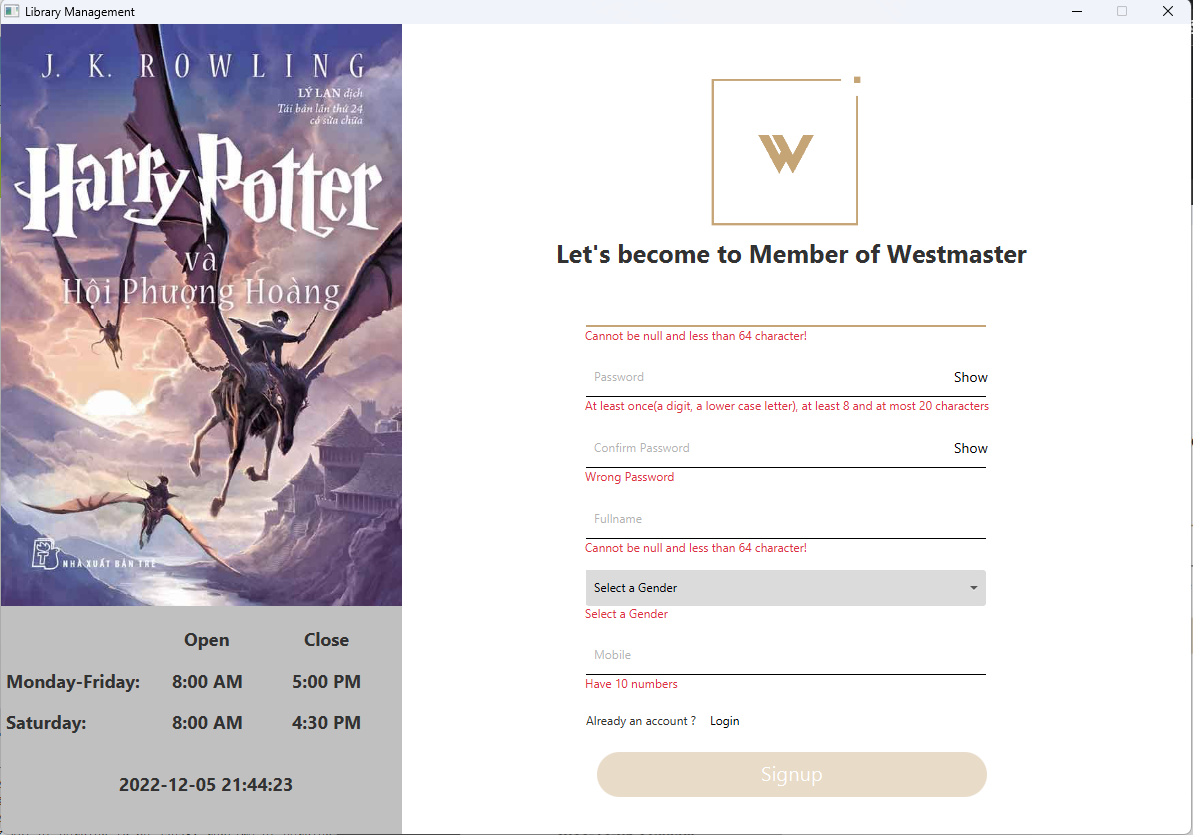
****

**Customer Profile**

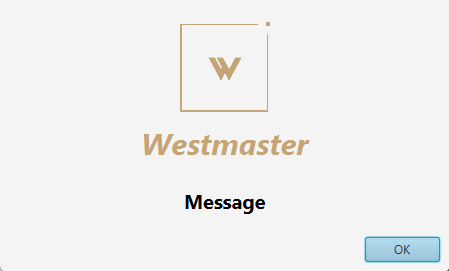
****

**Login Page**

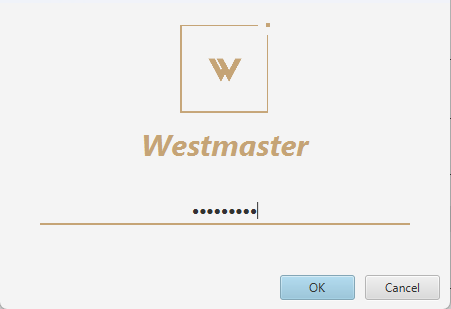
**SignUp Page**

****

**MessagePopUp**

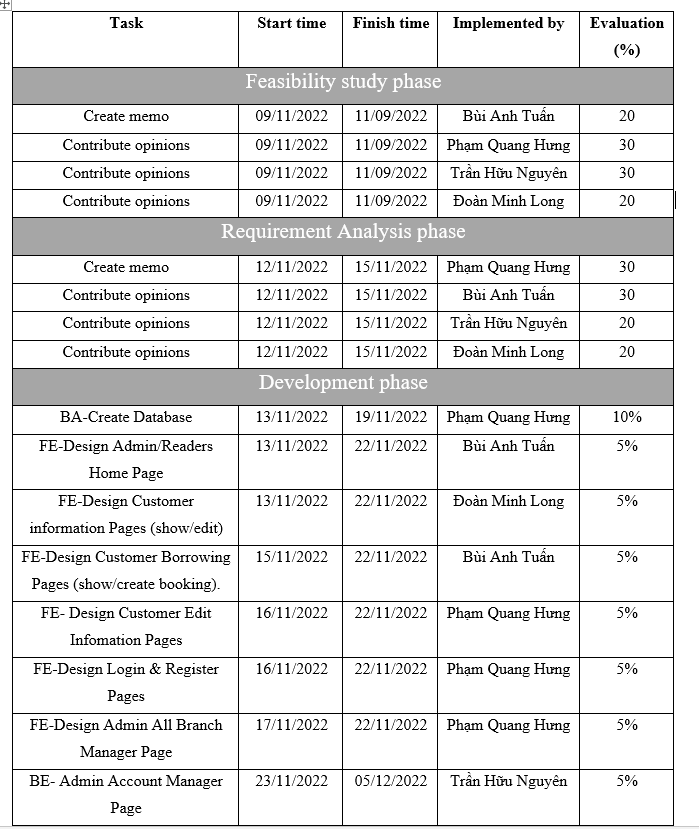
****

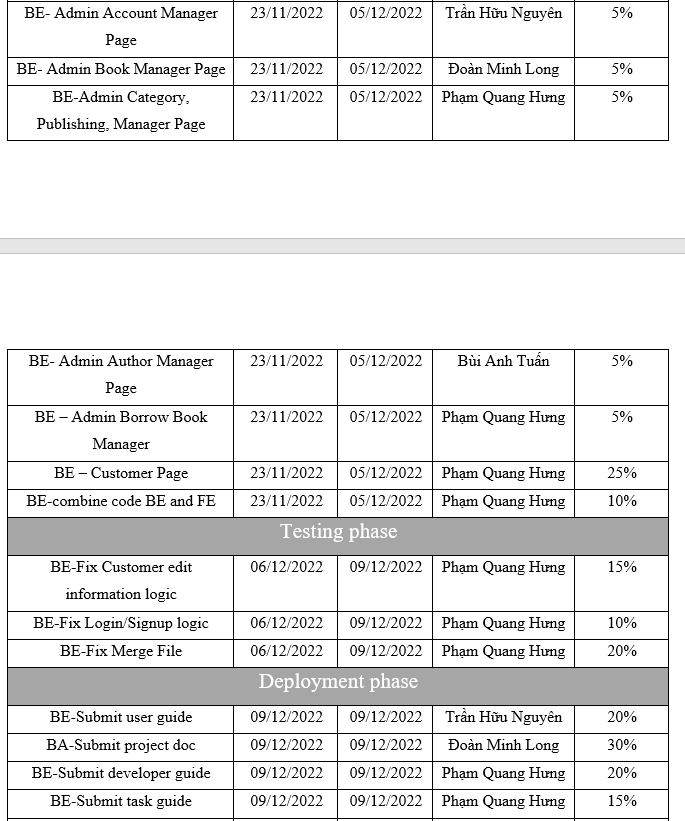
**ConfirmPassword PopUp**

****

**4.Validation Checklists**

| **Option** | **Validated** |
| --- | --- |
| **Do all links lead to the appropriate page?** | **Yes** |
| **Do all action links lead to the appropriate page?** | **Yes** |
| **Are all the images and links visible on the page?** | **Yes** |
| **Is the system taking too long to load?** | **No** |
| **Is the navigation sequence correct through all the pages on the system?** | **Yes** |
| **Is the page opening properly in all the tested tasks?** | **Yes** |

**5. Task sheet**

****

****

**6. Some algorithms applied in the project**

* **Sign-in:**

****

* **Search:**

****

* **Logout:**

****

* **Add new:**

****

* **Show information:**

****

* **Edit information:**

****

* **Delete:**

****