

Library Management Application*

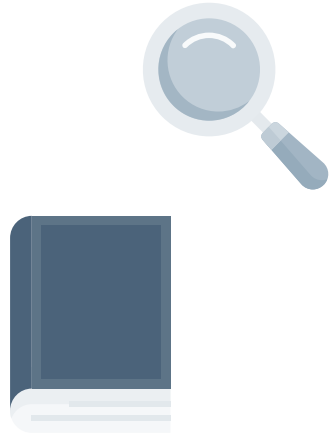
Jose Garcia-Esparza
Ricardo Sanchez-Macias

* Name TBD

System Description

Application with administrator users and end users

- For administrator, provide tasks to easily manage a library (e.g. adding, deleting, and editing books)
- For end user, provide tasks to easily utilize library resources (e.g. check-out books, search books)



Planned Implementation*

- Database
 - Supported by SQLAlchemy
- Backend
 - Python Flask
- Front-end
 - HTML and CSS



* Subject to change

Key Use Cases

For librarian and end user:

- To search books by different traits (author name, title, genre)



For end user:

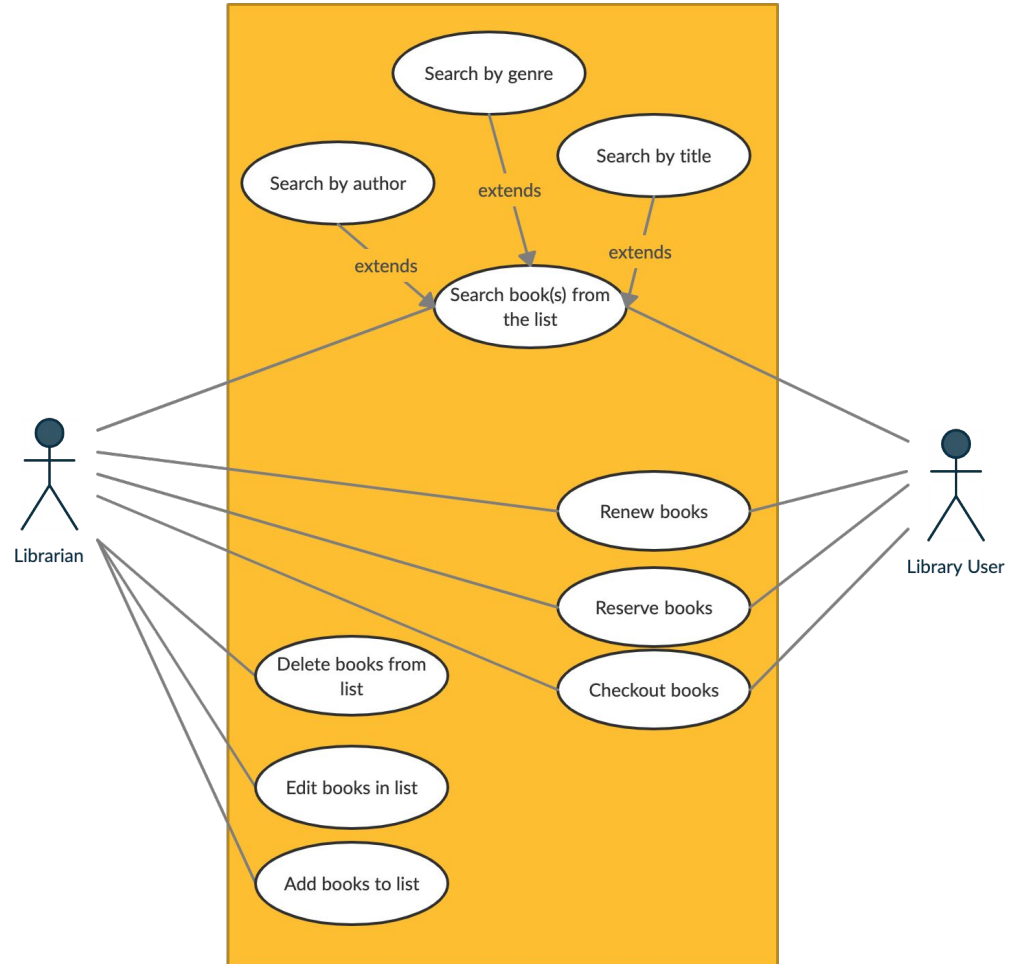
- Check-out books
- Reserve books if not available

For librarian:

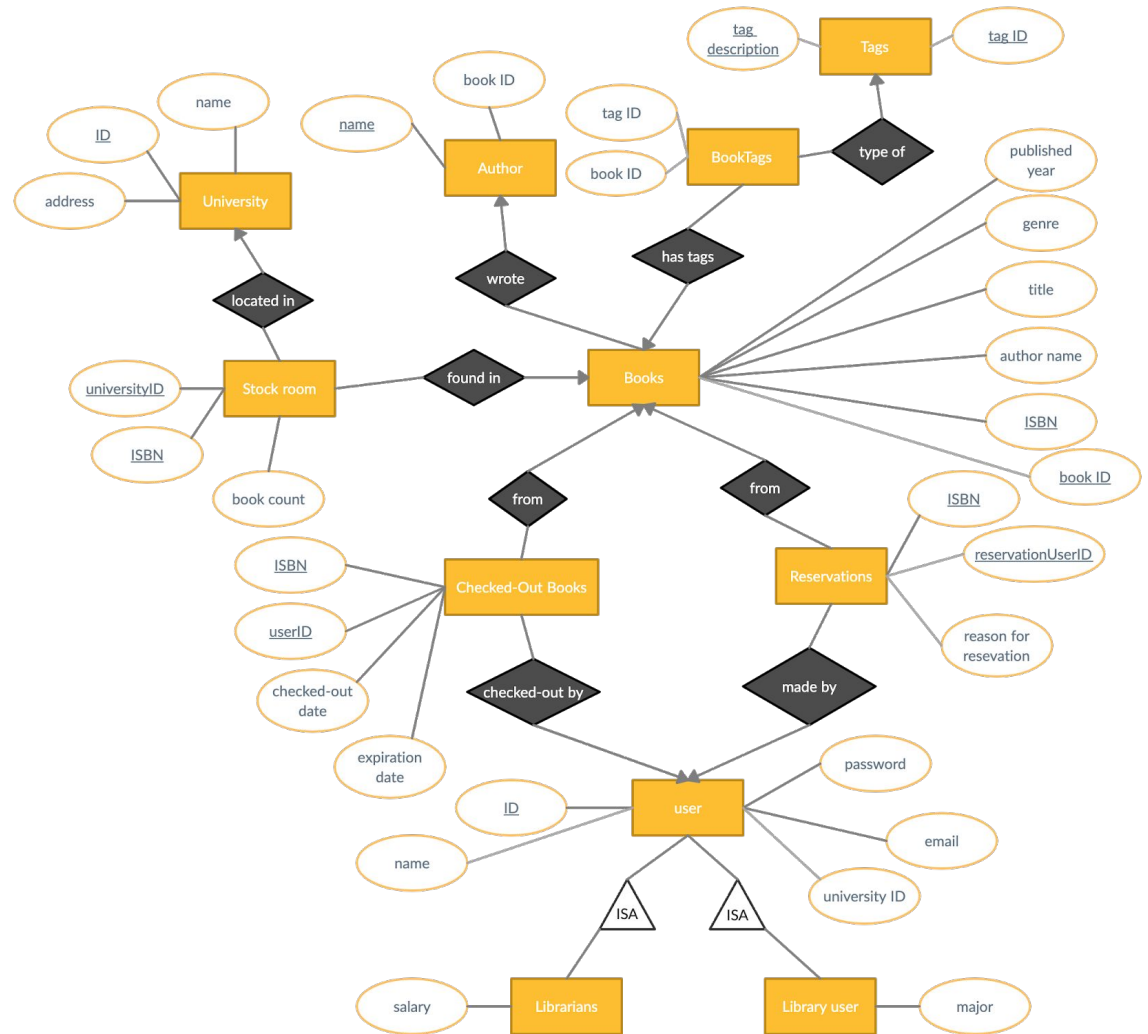
- Add, delete and edit books



User-Case Diagram



E/R Diagram



Relational Schema

Books:

- b_isbn, b_bookid, b_authurname, b_title, b_genre, b_publisheddate

Checked-out Books:

- cb_isbn, cb_userid, cb_checkeddate, cb_expirationdate

Reservations:

- r_isbn, r_fromuserid, r_reason

Users:

- u_userid, u_name, u_email, u_password, u_universityid

Librarians

- l_userid, l_salary

Library users:

- lu_userid, lu_major

Authors:

- a_name, a_isbn, a_address

Stock room:

- s_universidtyid, s_isbn, s_bookcount

University:

- uni_id, uni_name, uni_address

Tags:

- t_tagid, t_description

BookTags:

- bt_tagid, bt_bookid

Thanks!

Any questions?

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik** and illustrations by **Stories**