



Modern Application Development 2 - Project Report (May 2024)

By, Name: G.Ravi Teja, Rollno: 21f1005274, Email: 21f1005274@ds.study.iitm.ac.in

Selected Project: Library Management System V2

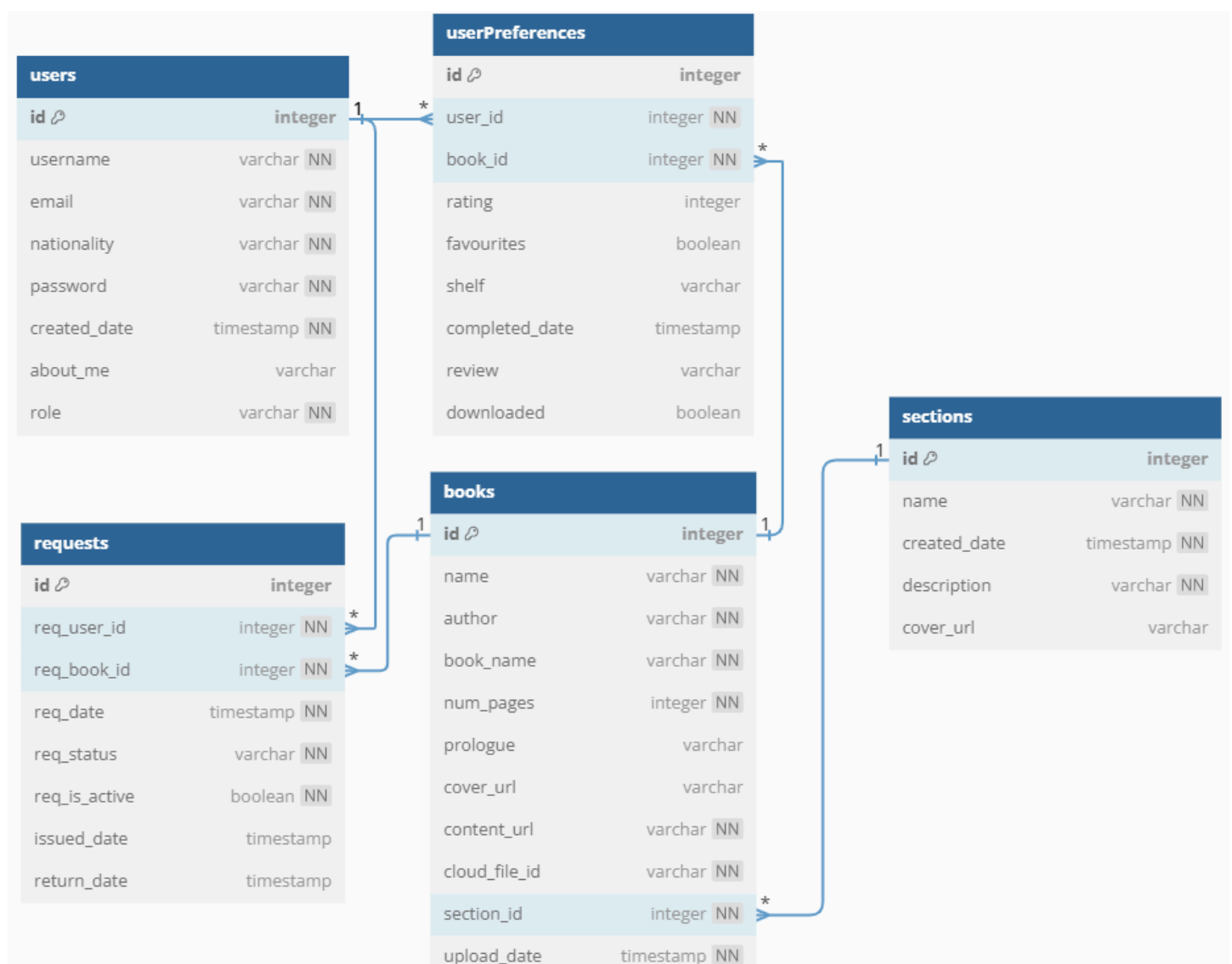
Problem Statement of the project:

Develop a multi-user app with two roles as librarian and user.

Librarian can monitor all the users, see all the statistics, create sections, e-books in the library.

A librarian can grant/revoke access for an e-book. General user can go through all the available sections and e-books in the library and can request for an e-book. A user can request for maximum of 5 e-books at a time. A user can update their profile page, rate an e-book.

ER Diagram of the Database:



Frameworks Used:

1. Flask - For Backend APIs
2. JWT Security - To protect the APIs using token based authentication and authorisation.
3. VueJS - For UI

4. SQLite - For Data Storage
5. SQLAlchemy - To manage database interactions in the APIs.
6. Google Drive - To Store the book pdf files, cover images.
7. Google Cloud Console - To create and manage Google Cloud projects, and Google APIs.
8. Service Account - To authenticate the app and grant permissions to access Google Drive.
9. Google Drive API - To programmatically upload files and manage permissions in Google Drive.
10. Redis - For Caching of the APIs for better performance.
11. Celery Integrated with Redis - For Sending Monthly Activity Reports, Daily Reminders.
12. Jinja2 Templates - For the Monthly Report template management.

Backend Architecture and API Design:

1. backend/app.py - Contains the main code to start the Library management system's server.
2. backend/model.py - Contains the Database Model for the project, code for admin creation.
3. backend/celery_config.py - Contains the setup for Celery Server, batch jobs schedules.
4. backend/tasks.py - Contains the code for batch jobs of monthly reports, daily reminders, requests csv generation.
5. backend/service_account.json - Contains the google drive authentication details for drive apis.
6. backend/report_template.html - Template for monthly report.

backend/routes

1. routes/__init__.py - Registers blueprints for all the APIs from all the files.
2. routes/cloud_file_management.py - Contains functions to upload file, cover to google drive, delete file, add email permission for a file, remove email permission for a file.
3. routes/auth.py - Contains User Register, Login, Update User Profile, Update User password, get all users information APIs.
4. routes/books.py - Contains CRUD Operation APIs for the books, APIs to get all the books information.
5. routes/sections.py - Contains CRUD Operation APIs for the sections.
6. routes/preferences.py - Contains APIs to add Reviews, ratings, favourites, shelves.
7. routes/stats.py - Contains APIs to generate the stats page data.

Librarian Features:

1. Librarian is created by default, Users can Register and Login to their accounts.
2. Librarian can Create a section, where cover image is optional. If the cover image is not provided, a default image will be shown for the section saying 'No Cover Available'.
3. Librarian can Update the name, description, and cover image of the sections.
4. Librarian can Delete the section in two possible ways:
 - 4.1 Delete the Sections, along with all the books.
 - 4.2 Move the Books of this section to any existing section and then Delete the section.

5. Librarian can Add a Book to any section, where cover image is optional. If the cover image is not provided a default image will be shown. The Book file will be uploaded to Google drives with permission as 'Viewers cannot download'.
6. Librarian can update all the information of the book at any time.
7. Librarian can move the book from one section to any other existing section.
8. Librarian can delete the book, which will also delete all it's related requests, preferences.
9. Librarian can view all the books, and also the Section specific books.
10. Librarian can Approve/Decline access permission to view the book for a user.
11. On Approval of a book access for a user, the User Email will be added to the book file in google drive with viewer permissions.
12. Once Approved, Librarian can Revoke the access permission for the book at any time.
13. A Celery Beat schedule runs daily to revoke the permissions of a book that are expired.
14. On Revoke/Return of a book, the User Email will be removed from the book file in drive.
15. Librarian can view the Book Specific Requests, User Specific Requests.
16. Librarian can view the Users public profiles.
17. Librarian can view the stats for the librarian management system.

Users Features:

18. Users can view all the sections, all the books, and Section specific books.
19. Users can Request for a book of their choice and can view the request status in My Requests.
20. Users can only request for 5 books at a time, users can withdraw a pending request.
21. Once Approved, Users can view the book, Users will be able to return the book within the expiry date. If not, returned, it will be auto revoked on the expiry date.
22. Once Approved, Users will be able to pay and Download the book.
23. Once Paid, Users will get the pdf downloaded to their local system, and also they will be able to view the book at any time in the Application without getting access revoked. **(Additional)**
24. Users can Give a Rating for the book.
25. Users can Write, View, Edit and Delete the Reviews for a book.
26. Users can View the Reviews of other users for any book.
27. Users Can add the book to shelves such as 'To Read Shelf', 'Currently Reading Shelf', 'Completed Shelf', 'Favourites'. **(Additional)**
28. Users can view the books from their shelves in the My Book Shelves Page. **(Additional)**
29. Users can view their profile, Update their Profile.
30. Users can view the stats of their most interested sections.

Video Link:

📎 21f1005274_LMS_Project_Demo.mkv