Library Management Application

Author

Yash Mishra
21f1006461@ds.study.iitm.ac.in

Video

https://drive.google.com/file/d/1ihIMQgZEJOAEUEwdQf85yIRldis8BDKY/view?usp=sharing

Description

The Library Management System is a modern solution for efficiently managing e-books and sections in a library environment. Developed using Flask for API and VueJS for the user interface, it offers essential functionalities for both librarians and users. Librarians can add, edit, and remove sections and e-books, while users can search for and request books. The system employs role-based access control for secure user authentication and features backend jobs for exporting, reporting, and sending emails daily and APIs for interaction with library resources.

Technologies used

- Python and following modules

- amqp==5.2.0
- aniso8601==9.0.1
- async-timeout==4.0.3
- bcrypt==3.2.2
- billiard==4.2.0
- blinker==1.5
- cachelib==0.9.0
- celery==5.3.5
- cffi==1.16.0
- chardet==5.2.0
- click==8.1.3
- click-didyoumean==0.3.0
- click-plugins==1.1.1
- click-repl==0.3.0
- colorama==0.4.6
- dnspython==2.3.0
- email-validator==1.3.1
- Flask==2.2.3
- Flask-Caching==2.1.0
- Flask-Excel==0.0.7
- Flask-Login==0.6.2
- Flask-Principal==0.4.0
- Flask-RESTful==0.3.10
- Flask-Security-Too==5.1.2
- Flask-SQLAlchemy==3.0.3
- Flask-WTF==1.1.1
- greenlet==2.0.2
- idna==3.4
- importlib-metadata==6.0.0
- itsdangerous==2.1.2
- Jinja2==3.1.2
- kombu==5.3.3
- lml==0.1.0
- MarkupSafe==2.1.2
- passlib==1.7.4
- prompt-toolkit==3.0.40
- pycparser==2.21
- pyexcel==0.7.0
- pyexcel-io==0.6.6
- pyexcel-webio==0.1.4
- python-dateutil==2.8.2
- pytz==2023.3.post1
- redis==5.0.1
- six==1.16.0
- SQLAlchemy==2.0.6
- texttable==1.7.0
- typing_extensions==4.5.0
- tzdata==2023.3
- vine==5.1.0
- wcwidth==0.2.9
- Werkzeug==2.2.3
- WTForms==3.0.1
- zipp==3.15.0

- Sqlite for database
- Flask for API
- Vue.js for UI
- Bootstrap for css
- Redis for caching
- Redis and Celery for batch jobs

DB Schema Design

Tables (5)

Name	Type	Schema
books		CREATE TABLE books (id INTEGER NOT NULL, title VARCHAR(50) NOT NULL, author VARCHAR(50), section VARCHAR(50), PRIMARY KEY (id))
id	INTEGER	"id" INTEGER NOT NULL
title	VARCHAR(50)	"title" VARCHAR(50) NOT NULL
author	VARCHAR(50)	"author" VARCHAR(50)
section	VARCHAR(50)	"section" VARCHAR(50)
issues		CREATE TABLE issues (id INTEGER NOT NULL, user_id VARCHAR, book_id INTEGER, r_datetime TIMESTAMP, action VARCHAR(15), a_datetime TIMESTAMP, PRIMARY KEY (id), FOREIGN KEY(user_id) REFERENCES user (email), FOREIGN KEY(book_id) REFERENCES books (id))
id	INTEGER	"id" INTEGER NOT NULL
user_id	VARCHAR	"user_id" VARCHAR
book_id	INTEGER	"book_id" INTEGER
r_datetime	TIMESTAMP	"r_datetime" TIMESTAMP
action	VARCHAR(15)	"action" VARCHAR(15)
a_datetime	TIMESTAMP	"a_datetime" TIMESTAMP
role		CREATE TABLE role (id INTEGER NOT NULL, name VARCHAR(80), description VARCHAR(255), PRIMARY KEY (id), UNIQUE (name))
id	INTEGER	"id" INTEGER NOT NULL
name	VARCHAR(80)	"name" VARCHAR(80)
description	VARCHAR(255)	"description" VARCHAR(255)
roles_users		CREATE TABLE roles_users (id INTEGER NOT NULL, user_id INTEGER, role_id INTEGER, PRIMARY KEY (id), FOREIGN KEY(user_id) REFERENCES user (id), FOREIGN KEY(role_id) REFERENCES role (id))
id	INTEGER	"id" INTEGER NOT NULL
user_id	INTEGER	"user_id" INTEGER
role_id	INTEGER	"role_id" INTEGER
user		CREATE TABLE user (id INTEGER NOT NULL, email VARCHAR, password VARCHAR(255), active BOOLEAN, fs_uniquifier VARCHAR(255) NOT NULL, PRIMARY KEY (id), UNIQUE (email), UNIQUE (fs_uniquifier))
id	INTEGER	"id" INTEGER NOT NULL
email	VARCHAR	"email" VARCHAR
password	VARCHAR(255)	"password" VARCHAR(255)
active	BOOLEAN	"active" BOOLEAN
fs_uniquifier	VARCHAR(255)	"fs_uniquifier" VARCHAR(255) NOT NULL