CSCE 608: Database Systems

Project Report

A Web Application for Paper Management

Ziyu Jiang

Project description

As a graduate student in CSE department, I need to read a lot of papers for my research. I find that it is hard to manage them by only put them into the folder of my computer. It is hard to find the paper you want by only searching the name of papers. In order to manage them in a more convenient and straightforward way, I designed this paper management system. There are some subsystems for my design:

- Authentication System
    - Handles user accounts registrations.
- Paper Category Management System
    - Allow user to create edit and delete the category
- Paper Management System
    - Allow user to create edit and delete the paper
    - Allow user to search papers by keywords in title and abstract and category of the paper

Authentication System

Figure 1. Register Page

Authentication system offers functions like registration and logging in. User could register an account and log in to manage their own papers. The view of registration page is shown in Figure 1.

Paper Category Management System

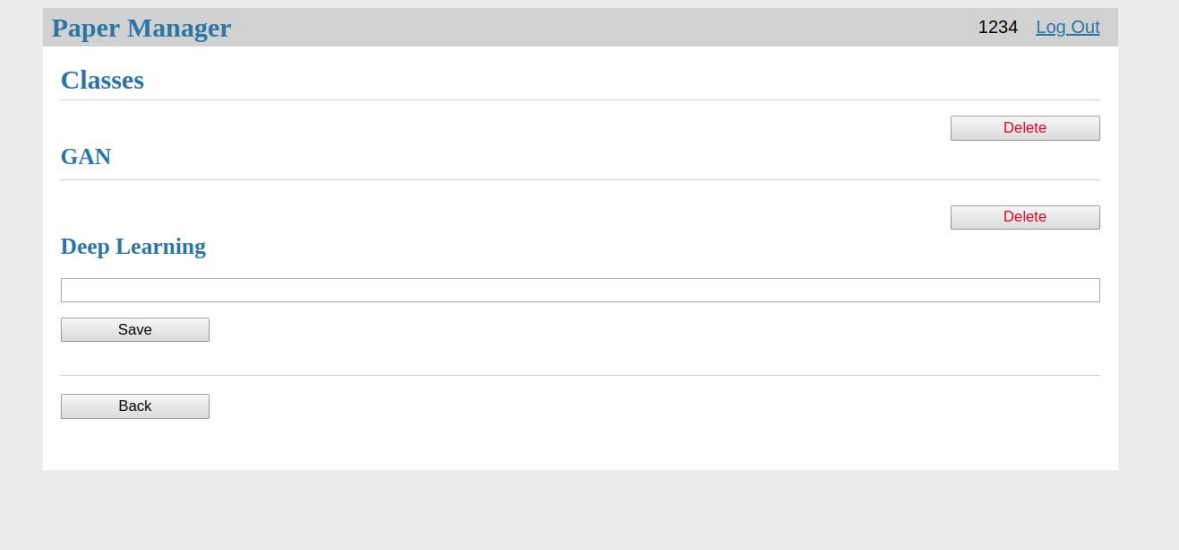

Figure 2. Paper Category Management System Page

Paper category management System allows user to manager the categories/classes of the paper by adding or deleting. The view of the management system is shown in Figure 2.

Paper Management System

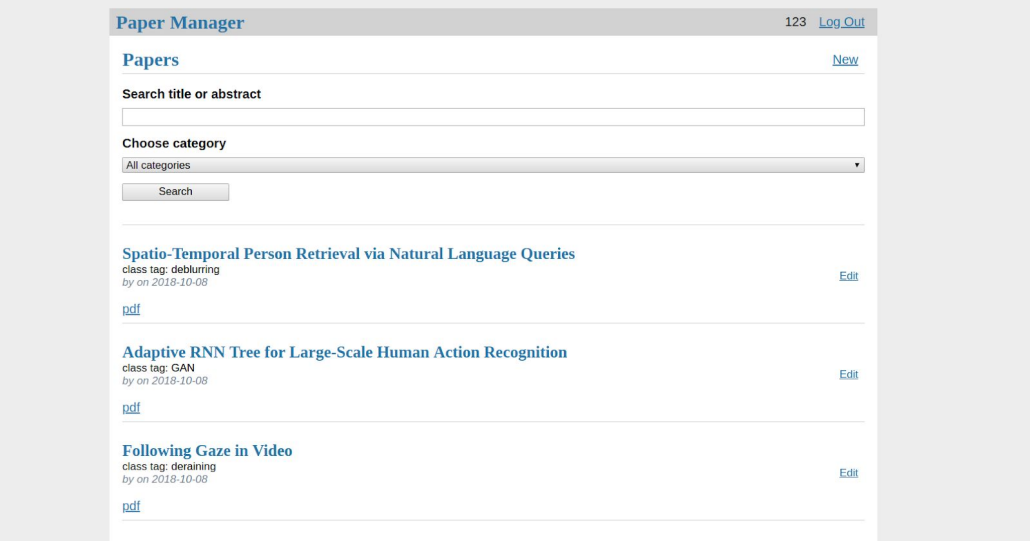

Figure 3. Paper Index Page

Users can create, edit and delete papers in Paper Management System. They can save the title, abstract and link of the paper. They can also search their papers by category of the paper and keywords in title and abstract.

Data Collection

We collect the data from open access paper websites of two conferences. One is ICCV, the other is CVPR. The url of the websites is as follows

http://openaccess.thecvf.com/CVPR2018.py

http://openaccess.thecvf.com/ICCV2017.py

It is worth nothing to mention that we only collect title and link of papers from these websites. Cause it's difficult to collect abstract. We collect title and abstract by reading these websites by python and then extract the information with regular expression technics. We collect 1134 papers in total. They be viewed after logging in with username '123' and password '123'.

Entity-Relationship Diagram

The entity-relationship diagram is shown in figure 4:

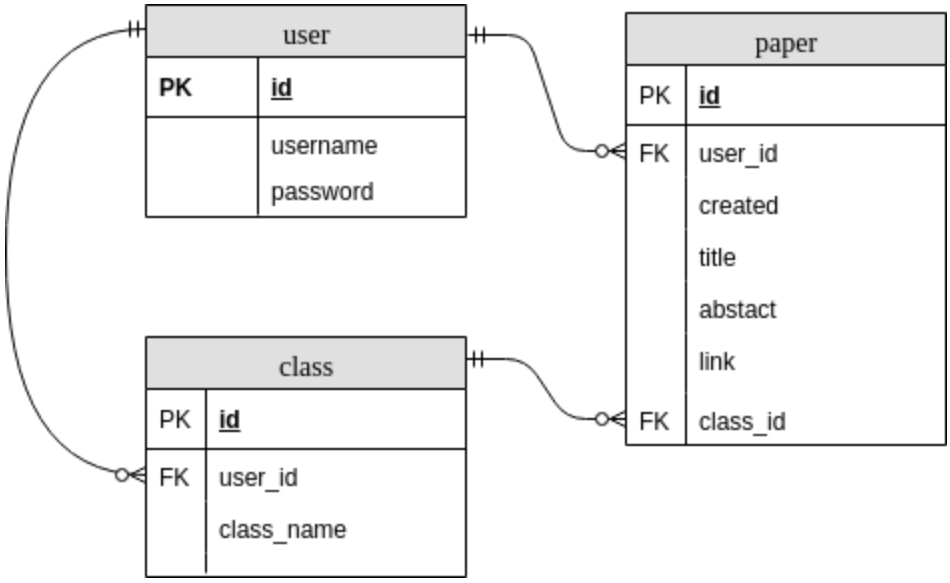

Figure 4. entity-relationship diagram

Table normalization

These tables have been Boyce-Codd normal form. So table normalization cannot improve the structure further. In user table, the id is the super key and all other attributes depends on the id. In class table, the id is the super key and all other attributes depends on the id. In paper table, the id is the super key and all other attributes depends on the id. So, for any dependency A → B, A is a super key.

User interface

I designed the user interface via adding css file in static directory.

As a user, when he open the the website in https://papermanager.herokuapp.com/, he would see the log in page. If he is a new user, he could click the Register link in the right top corner to create a new account. After registration, he would be directed to “log in” page and he can log in with his account.

After logging in, he can see the index page of the papers he collected. He could create a new paper by clicking the right top link “New”. In the creating page, he could manage the categories of paper by clicking “Manage Classes”.

If user want to edit or delete the paper he create, he could click the “edit” link in the end of the line of paper. In edit page, he could edit the title, abstract and class of the paper. There is also a “Delete” link in the bottom of the page. By pressing it, the user could delete this paper.

If user want to search the papers. He could do it in the paper index page. By choosing category beyond the search button and pressing the search button. Only papers in that categories would be displayed. If he want to search it by keywords. He could easily do it by enter the keywords in the search input area and press search. Only papers with that keywords in title or abstract would be displayed.

Project Source Code

Project source code is uploaded to github:

https://github.com/geekJZY/try_Flask

Discussion

The biggest problem I’ve encountered is in deploying. At first, I cannot deploy the project to heroku because heroku doesn’t support the database type I used. Then I switched from the sqllite to postgresql. It finally works.