

计算机网络与Web技术大作业报告

小组成员：朱熠恺 1600012810，吴睿海 1600012824，徐沁玫 1600012973，赵宇瑄 1600012984

计算机网络与Web技术大作业报告

基本介绍

使用环境

使用说明及示例

用户

注册

登录

个人主页&个人设置

帖子

发帖

回复

文件

管理员

用户管理

帖子管理

文件管理

组内分工

后端：

前端：

源程序整理

数据库

MySQL

user 用户

post 帖子

post_file 文件

reply 回复

collects 收藏

likes 点赞

功能部分

用户管理

注册

登录

验证码

个人主页

个人设置

帖子管理

主界面

热贴

发贴

文件

回复

搜索功能

管理员界面

管理员首页

用户管理
帖子管理
文件管理
页面美化
轮播图
背景画布
logo设计

基本介绍

本次大作业实现了一个社交网路的实现，我们将其命名为ICS论坛，可以实现登录，注册，发帖，回复，点赞，收藏，文件传输，管理员管理用户、帖子和文件等功能，同时还有较为美观的背景。

[GitHub源码地址](#)

使用环境

使用anaconda3.6，所需要的库要求均放在requirements.txt中。

首先需要安装MySQL 8.0.12

然后运行 `pip install -r requirement.txt`

然后进入flaskr文件夹，执行如下语句

```
venv\Scripts\activate
set FLASK_ENV=development
set ELASK_DEBUG=1
set FLASK_APP=flaskr
flask run
```

使用说明及示例

用户

注册

1. 点击首页左上角的注册按钮进入注册界面。
2. 依次填写用户名、昵称、邮箱、密码、验证码。其中密码需要连续输入两次进行确认。
3. 若填写均无误，点击提交，即可注册完毕。
4. 若看不清验证码，点击图片即可更新

登入

注册

用户名

将会成为您唯一的登入名

昵称

邮箱

请注意邮箱格式

密码

6到16个字符

确认密码

验证码

请输入验证码



提交

登录

1. 点击首页左上角的登录按钮进入登录界面，注册完毕之后也会自动跳转到登录界面
2. 输入用户名，密码，验证码
3. 若用户名已被注册且密码正确，则进入主界面，同时主界面中显示
4. 若看不清验证码，点击图片即可更新

说明：

- 如果在数据库中没有找到匹配的用户名，则回到登录界面并显示“用户名不存在”。
- 如果输入的验证码错误，回到登录界面并显示“验证码错误”。

登入

注册

用户名


zykzyk

密码

.....

验证码

请输入验证码



立即登录

个人主页&个人设置

点击对应按钮进入界面即可。



zwk
🕒 2018-12-23 加入

zwk 最近发布的帖子

zyk	2018-12-28	收藏/回复
test_file	2018-12-25	收藏/回复
miao	2018-12-25	收藏/回复
test_file	2018-12-25	收藏/回复
test_file	2018-12-25	收藏/回复

zwk 最近收藏的帖子

zyk	2018-12-28	收藏/回复
-----	------------	-------

帖子

发帖

点击 创建新帖 按钮，填写帖子的主题、内容部分即可发帖。

发表新帖

标题

B U ↻ | 🔗 😊

立即发布

选择文件

未选择任何文件

回复

进入帖子界面，填写回复内容点击回复即可。

回帖

B U  |  

提交回复

文件

- 上传步骤：用户在发帖时，若想上传文件，可以点击发表新帖页面中的选择文件，选择想要上传的本地文件。本次仅允许用户上传一个文件，多个文件可以通过将其压缩的方式方式上传。
- 下载步骤：用户想要下载某文件，可以在帖子展示界面点击显示有文件的按钮，如果文件安全，用户就可以选择希望文件下载到的用户电脑上的地址，之后就可以开始文件传输。

zyk

取消收藏 点赞



zykzyk 2018-12-28


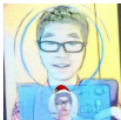


删除此帖

zyk——file

HW7.md

管理员

开发者

	朱熠恺	1600012810
	吴睿海	1600012824
	徐沁孜	1600012973
	赵宇澄	1600012984

用户管理

1. 在管理员首页点击 用户管理 按钮可以进入用户管理界面。
2. 在用户管理界面可以显示所有用户的ID，昵称，创建时间、邮箱和禁言状态。
3. 若想搜索对应用户的信息，可以在搜索框输入目标用户的用户名或者昵称的信息，点击搜索用户，即可显示对应用户的相关信息。关键词没有必要是完全匹配，可以是部分匹配。
4. 点击 禁言 按钮即可完成对相应用户禁言的功能。用户一旦被禁言，在管理员界面禁言状态显示 已禁言，在论坛对应的界面中， 发表新帖 和 回复 按钮就会消失，如此该用户就无法发帖、无法回复。
5. 点击 删除 按钮即可完成删除用户，管理员界面不再显示该用户，用户无法登陆论坛并发帖。

管理员首页 用户管理 帖子管理 文件管理						
请输入用户名称						搜索用户
用户ID	用户昵称	注册时间	邮箱	禁言状态	禁言操作	删除操作
4	zwwk	2018-12-23 11:37:58		未被禁言	禁言	删除用户
5	123	2018-12-23 11:51:44		未被禁言	禁言	删除用户
6	nick1	2018-12-23 14:16:20		未被禁言	禁言	删除用户
7	nick2	2018-12-23 14:17:48	skjadkid@gmail.com	已被禁言	解除禁言	删除用户
10	nickzyk	2018-12-23 18:48:18	skjadkid@gmail.com	未被禁言	禁言	删除用户

帖子管理

1. 在管理员首页点击 帖子管理 按钮可以进入帖子管理界面。
2. 在帖子管理界面可以显示所有帖子的发帖用户、发帖题目、发帖时间、置顶状态等信息，
3. 若想搜索对应帖子的信息，可以在搜索框输入目标帖子的标题、内容相关的信息和置顶状态，点击搜索帖子，即可显示对应帖子的相关信息。关键词没有必要是完全匹配，可以是部分匹配。
4. 点击 置顶、删除帖子 即可完成置顶帖子和删除帖子相关操作。一旦帖子被置顶，则在主页展示的位置会位于未置顶的帖子之前，置顶的帖子会根据时间顺序决定谁在前；删除帖子后，与该帖子相关的回复、文件等信息全部被删除。

文件管理

首页

管理员首页

用户管理

帖子管理

文件管理

请输入帖子相关信息

搜索帖子

帖子ID	用户昵称	发帖题目	发帖时间	置顶状态	置顶操作	删除
2	123	123	2018-12-23 11:52:01	未被置顶	<div>置顶</div>	<div>删除帖子</div>
4	zwk	asdklajklsa	2018-12-23 18:49:40	未被置顶	<div>置顶</div>	<div>删除帖子</div>
5	zwk	sadjald	2018-12-23 18:49:54	未被置顶	<div>置顶</div>	<div>删除帖子</div>
6	zwk	testfile	2018-12-25 20:43:27	未被置顶	<div>置顶</div>	<div>删除帖子</div>
7	zwk	testfile	2018-12-25 20:45:17	未被置顶	<div>置顶</div>	<div>删除帖子</div>
8	zwk	testfile	2018-12-25 20:45:51	未被置顶	<div>置顶</div>	<div>删除帖子</div>

- 1. 在管理员首页点击 [文件管理](#) 按钮可以进入文件管理界面。
- 2. 在帖子管理界面可以显示所有帖子的上传用户id、上传用户昵称、文件名等信息
- 3. 若想搜索对应文件的信息，可以在搜索框输入目标文件的标题相关的信息，点击搜索帖子，即可显示对应文件的相关信息。关键词没有必要是完全匹配，可以是部分匹配。
- 4. 点击 [删除文件](#)，与该文件相关的表项就会被对应删除。

[首页](#)

[管理员首页](#)

[用户管理](#)

[帖子管理](#)

[文件管理](#)

请输入文件名

搜索文件

上传用户ID	用户昵称	文件名	删除
5	123	1.png	删除文件
4	zwk	1.py	删除文件
4	zwk	13.md	删除文件
4	zwk	HW4.md	删除文件
4	zwk	HW7.md	删除文件

组内分工

后端：

朱熠恺：注册、登录、文件传输、用户管理、帖子管理、文件管理、管理员界面的搜索功能

吴睿海：验证码、用户个人主页、用户个人设置、热帖、回复、用户界面的搜索功能

前端：

徐沁玫：注册、登录、文件传输接口、用户个人主页、个人设置、主界面、发帖

赵宇瑄：管理员首页，用户管理、帖子管理、文件管理、轮播图、背景画布、logo设计

源程序整理

数据库

MySQL

MySQL是小型关系型数据库管理系统，由瑞典MySQL AB公司开发，主要应用于网站后台，支持FreeBSD、Linux、MAC、Windows等多种操作系统。MySQL拥有大型商用DBMS的大多数特性，如事务、子查询、触发器、试图、锁定等，可以处理拥有上千万条记录的大型数据，支持常见的SQL语句规范，可移植行高，有丰富信息的网络支持。同时它还具有一些其他数据库管理系统所没有的特性，比如用户定义类型、继承、规则和多版本并行控制等，相比其他大型数据库，调试、管理、优化相对简单。

在前期经过调研之后，我们决定使用 `MySQL` 。

PEEWEE

使用Python ORM Peewee来操作数据库

对象关系映射（Object Relational Mapping，简称ORM），是一种程序设计技术，用于实现面向对象编程语言里不同类型系统的数据之间的转换。对象和关系数据是业务实体的两种表现形式，业务实体在内存中表现为对象，在数据库中表现为关系数据。内存中的对象之间存在关联和继承关系，而在数据库中，关系数据无法直接表达多对多关联和继承关系。因此，对象-关系映射(ORM)系统一般以中间件的形式存在，主要实现程序对象到关系数据库数据的映射。

peewee是一个轻量级的ORM，基于SQLAlchemy内核，采用纯python编写，显得十分轻便，很容易和任意web框架集成。且peewee是Django式的API，简单易用，容易学习，因此我们决定使用peewee。

接下来对数据库中各个TABLE逐一进行说明。

user 用户

id 用户编号

created 创建时间

username 用户名

nickname 昵称

password 密码

email 邮箱

is_block 是否被禁言

post 帖子

id 帖子编号

created 创建时间

author_id 作者编号

num_view 点击量

num_reply 回复数

num_like 点赞数

num_collect 收藏数

hot 热门指数

title 帖子标题

body 帖子内容

is_top 是否置顶

is_fine 是否精品

post_file 文件

id 文件编号

post_id 帖子编号

created 创建时间

filename 文件名

filehash 文件哈希值

reply 回复

id 回复编号

author_id 作者编号

post_id 帖子编号

created 创建时间

body 回复内容

collects 收藏

id 收藏编号

author_id 作者编号

post_id 帖子编号

likes 点赞

id 点赞编号

author_id 作者编号

post_id 帖子编号

功能部分

本次后端主要使用 `python`，以 `flask` 作为Web应用框架。实现了社交网络的基本功能。前端主要使用了html、javascript，使用layui模板对贴吧网页进行搭建。

Flask 是一个 web 微框架，不依赖于外部库的框架，自由灵活轻量级，可扩展性强，更新时依赖少，并且专注安全方面的 bug。Flask对于选择何种数据库、使用何种模板引擎不做要求，因此易于开发和学习，满足我们项目的各项要求。Flask 的依赖包括Werkzeug 一个 WSGI 工具包和jinja2 模板引擎。模板引擎可以帮助设置页面的基本布局，在局部元素发生变化时，可以保持其他页面布局不变，仅进行部分修改，以免造成代码冗余，在创建、更新、维护应用时节约时间。

本项目使用layui的html模板。目前网络上有很多非常实用且漂亮的网页模板，它们都是基于html和javascript语言写成的。我们选择layui的原因是尽管它的功能没有其他一些模板强大，但是也足够我们使用。layui的模板文档和封装都写得很好，非常便于我们快速学习上手。layui模板同时提供了一个静态网页版的贴吧可以供我们在设计前端版式时参考，使我们能将更多的精力放在研究前后端接口等问题上。

用户管理

注册

前端使用layui模板来构造贴吧结构，layui模板提供了封装好的提交表单功能，因此注册信息可以方便地使用表单提交。

用户输入的所有用户信息组成表单使用post方法向后端发送请求，表单就会提交到后端 `register()`，只要将每个信息的id合理命名，后端就可以顺利收到表单并针对每个不同的信息进行不同的处理。

同时前端也使用verify在表单提交到后台前对表单输入内容的格式进行检查。比方说必填项不为空、邮箱、用户名、密码格式需要复合既定格式，如果不符合要求就会弹出警告并且不会把表单提交到后台。

后端利用 `get_register_info()` 函数，从 `request` 中解析出用户输入的 `username`，`password`，`nickname`，`email`，`imagecode`。如果一切都是合法的，那么在数据库中的user表总插入对应表项，并自动转到登录界面。其中若未输入昵称，则默认昵称就是用户名。

若不合法，则回到注册界面并显示响应错误提示。可能出现的错误有：

1. 未输入用户名、密码、电子邮件
2. 用户名和已经注册用户重名
3. 两次输入的密码不匹配
4. 验证码错误
5. 用户名或昵称过长(超过40个字符)

这里不直接存取 `password`，而是使用 `werkzeug.security` 中的 `generate_password_hash` 函数，计算出 `password` 的哈希值，最后将该哈希值存储在数据库中。如此就算数据库中的数据出现泄漏，也可以保障用户的密码信息这个关键信息不被泄漏。

登录

用户输入的所有登录信息组成表单使用post方法提交到后台，只要将每个信息的id合理命名，后端就可以顺利收到表单并每个不同的信息进行不同的处理。

前端也实现了对表单输入内容格式的检查，比方说必填项不为空等，如果不符合要求就会弹出警告并且不会把表单提交到后台。

由于在数据库中存储的是注册时使用密码的哈希值，因此登录时检查密码是否匹配的过程具体如下：利用同样的哈希函数为输入的密码生成哈希值，若此哈希值和数据库中存储的哈希值相同，则认为输入的密码是正确的。否则，回到登录界面并显示“密码不正确”。

验证码

前端用jinja2接收后端生成的验证码，并将其直接展示在页面上。

后端生成、验准验证码。

`valid_picture()` 函数生成五个字符的验证码，拥有大小、位置、形状的改变，有干扰线和模糊滤镜。

`get_code()` 函数将验证码图片以二进制形式写入在内存中，防止将图片都放在文件夹中、占用磁盘空间。把这个二进制形式的验证码作为response返回给前端，并在 `session['imagecode']` 中记录验证码对应的字符串。

如果验证码图片看不清，点击验证码即可刷新，因为点击一次验证码图片即调用一次 `get_code()` 函数。

在注册和登陆时，验准用户输入的验证码和 `session['imagecode']` 中记录验证码字符串是否相同。

个人主页

前端获取后端返回的用户信息、用户发过的贴子和用户收藏的贴子并展示到页面上。

后端利用 `get_home_info()` 函数，分别在 `user table`、`posts table` 和 `collects table` 中获取用户信息、用户发过的贴子和用户收藏的贴子。

个人设置

前端通过使用post方法向后端发送request的方式提交表单，将用户修改的信息传向后端 `set_name()`、`set_mail()` 和 `set_pass()` 三个函数里。

后端利用 `set_name()`、`set_mail()` 和 `set_pass()` 三个函数，分别通过向数据库写入的方式，更改用户昵称、邮箱和密码。

贴子管理

主界面

主界面中后端会给前端所有帖子的信息，包括每个帖子的名称、发帖者、发帖时间、点赞数、回复数、收藏数、是否置顶、是否是热贴。前端根据这些信息将帖子罗列在主界面上。

从 `post table` 中获取贴子、点赞数、回复数、收藏数、是否置顶等信息。

从 `user table` 中获取对应贴子的作者信息。

从 `post_file table` 中获取对应贴子的附件情况。

从 `post table` 中以 `hot` 字段（由计算热度的线程定时计算）降序获取热贴。

热贴

前端通过接收后端传来的hots获得所有热帖的信息。

热贴计算使用 `Hacker News` 的方法： $Score = \frac{numreply+numview+1}{(\delta+2)^{1.8}}$ 其中，`numreply`是回复数，`numview`是发帖数，`delta`是发帖到现在的时间，以小时为单位。

为了实时计算热贴，需要新开一个线程，隔一段时间唤醒一次，更新数据库中每个贴子的 `score`。具体实现请参见 `start_calc.py` 和 `hot_calculator.py` 两个程序，分别是线程管理程序和热度计算程序。

发帖

前端通过使用post方法向后端发送request的方式提交表单，将用户修改的信息传向后端 `create()` 里。

发帖的内容编辑部分使用了layui包装的富文本编辑器，使得帖子的内容和形式可以更加丰富。layui的富文本编辑器能够实现一些排版功能。我们的贴吧采用其功能中的加粗、下划线、划去、分割线、超链接和表情。由于富文本编辑器不是html本身携带的表单样式，因此如果要提交表单的话需要多几步处理。一般创建富文本编辑器是先创建 `<textarea>`，再在 `<textarea>` 里“画出”富文本编辑器样式。因此在富文本编辑器内输入编辑的内容之后，如果要提交，首先要把富文本编辑器的内容同步到 `<textarea>` 里，接着使用post提交才能将富文本编辑器内的内容提交到后端。这里同步使用了javascript自定义了判断函数，在判断函数中写了同步操作。需要注意的是有可能在同步结束之前内容已经被提交导致内容不能正常提交到后端，因此设置了`sleep(100)`使得同步操作有充分的时间执行。

发帖时，先确认前端输入了 `title`。

之后向 `post table` 写入 `title`、`body`、`author_id` 等信息，默认不加精、不置顶，点赞数、收藏数、访问量初始值为0。

如果上传了文件，在 `post_file table` 中写入文件。

文件

我们并不在数据库中存取文件的全部内容，而是只是在数据库中维护文件的文件名和一个哈希值，而将文件存在服务器本地一个文件夹。这样做是考虑到文件的大小可能会有很大的差异，如果在数据库里维护添加一系列来存放文件内容，不同文件的这一信息就会有很大差异，很难管理。同时一旦有一个很大的文件，就会导致整个数据库的内容迅速变大，使得在数据库上的操作变得极其缓慢。

为了解决存在服务器本地的文件安全性问题，我们选择在数据库中存取文件的一个hash值，以此记录文件最开始的信息。之后用户需要下载文件的时候，我们通过这个哈希值检查此时存在服务器本地的文件是否和最初存入时内容相同。一旦内容被改动过，我们就拒绝用户下载这个文件，表明这个文件可能已经不再安全。

基于此，前端需要提供上传文件和下载文件的接口。目前的设计是：上传文件动作只能在创建帖子时操作；而下载文件是在查看某一帖子详细信息的页面上实现。如果某个帖子附加了文件，那么在帖子正文下方会出现一个名字为文件名称的按钮，点下按钮即可下载。如果出现异常状况会报错。

用户想要上传文件时，可以将本地文件的文件内容和文件名等信息通过POST的方式传给服务器后端，然后我们通过哈希函数记录此时文件内容的哈希值，将该哈希值和文件名一起存入数据库的 `post_file` 表格中。

下载时，后端利用数据库中的信息检查文件是否被改动，一旦被改动，则返回显示帖子的界面，并显示错误信息“文件已被改动”来暗示用户这个文件已经不再安全，不能下载这个文件。

回复

回复与创建新帖子同样使用了layui模板的富文本编辑器，实现起来与创建新帖子基本相同。

添加回复时，从前端 `body`、`author_id`、`post_id` 添加进 `reply table`。

删除回复时，从 `reply table` 中删除 `reply_id` 匹配的内容。

注意添加或删除回复的同时，要更新 `post table` 里对应贴子的 `num_reply` 字段。

搜索功能

搜索功能也是通过提交表单实现向后端传递关键词，一开始想将值传进一个专门的函数SEARCH_TITLE里，但是没有成功。最后实现的办法是将值传进各页面对应的函数中去。

使用 `like()` 语句进行匹配。

要搜索关键词是ST，就在 `like()` 中的关键词前后要加通配符 `%`。

具体可见 `title_search()` 函数和 `user_search()` 函数。

管理员界面

如果检测到登录者是管理员，则在主界面会多一个按钮：`管理界面`。管理员可以通过此按钮进入管理员首页。本地代码默认管理员只有一个，且最先注册的人就是管理员。在管理界面的左上方有5个按钮，通过这些按钮分别可以进入对应的首页和管理界面。

管理员首页

管理员首页可以看到开发者信息和开发记录日志

用户管理

前端通过使用post方法向后端发送request的方式提交表单，将搜索的用户的信息传向后端 `admin.SearchMember` 里，由后端返回搜索用户信息。

想要禁言或删除用户时，前端将用户相关信息发送给后端；后端接收到接收到前端发来的相关信息，根据这个信息修改数据库中对应表项即可完成修改。

删除用户时，首先找到和这个用户相关的所有帖子和回复，先将它们全部删除，之后在把数据库中对应表项删除。

帖子管理

前端通过使用post方法向后端发送request的方式提交表单，将搜索的帖子的信息传向后端 `admin.SearchPost` 里，由后端返回搜索用户信息。

想要置顶帖子时，前端将帖子的id等信息传给后端，后端根据这个信息修改数据库中对应的表项。

想要删除帖子时，前端将帖子的id等信息传给后端，后端先在数据库中找到所有关于相关的回复、文件、收藏、喜欢，先将它们删除之后，再将原帖子对应的表项删除。

文件管理

前端通过使用post方法向后端发送request的方式提交表单，将搜索的文件的信息传向后端 `admin.SearchDoc` 里，由后端返回搜索用户信息。

想要删除文件时，前端将文件的id等信息传给后端，后端根据这个信息删除数据库中对应的表项。

页面美化

轮播图

前端利用layui模板中的 `carousel` 模块加载轮播图，通过模块内javascript脚本实现图片自动滚动及按钮翻页功能。

背景画布

前端利用html5中的canvas画布进行论坛整体动态背景绘制，通过javascript脚本 `ctx.fillStyle` 实现canvas上的字符随机飘落效果。

通过html属性 `style="position: absolute` 固定各模块位置，实现画布处于整体布局最下层的设计。

logo设计

纸飞机象征着远航和希望，预示着通过论坛共享知识资料、解答学科难题，用户可以在计算机学科中更进一步。

ICS既是Introduction to Computer Science，也是爱CS，希望用户可以共同维护社区和谐，充分利用社区资源。