

课程设计报告

( 2019--2020年度第2学期 )

课程名称：python程序设计与数据分析

题 目： 开发设计留言板系统

院 系： 控制与计算机工程学院

班 级： 软件1801

学 号： 120181080226

学生姓名： 孙志岩

指导教师： 熊建国

设计周数： 1周

成 绩：

日期： 2020 年 6 月 29日

目录

[一．功能概述 3](#_Toc43923827)

[二、所使用关键python库 3](#_Toc43923828)

[三、数据库表的设计 4](#_Toc43923829)

[四、文件结构 4](#_Toc43923830)

[五、功能详细介绍 5](#_Toc43923830)

## 一．功能概述

使用python语言设计一个基于flask框架留言板系统；用户注册登陆后，能发布留言，分别把每一条留言信息按照[ID，作者ID，信息内容，发布时间]存储到message数据库中，本人能对自己发布的留言进行删除和修改管理；能查看留言列表，分页显示，能根据留言的内容，做模糊查询；

## 二、依赖的python模块

1.flask ：应用框架

2.jinja2 ： 基于python的模块引擎

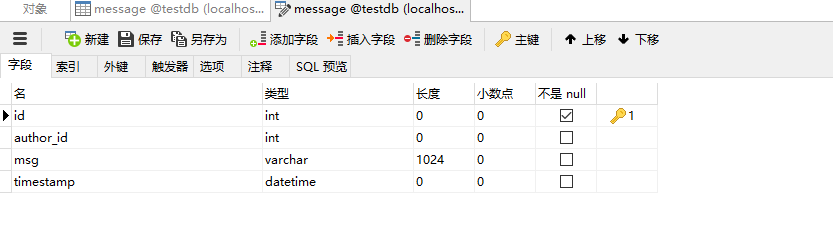
3.bootstrap ： 留言板界面主要继承bootstrap/base.html模板

4.flask-login ： 会话保持

5.SQLAlchemy ：提供python连接，使用，修改message数据库的函数

6.Flask-script ：提供向Flask插入外部脚本的功能

## 三、数据库表的设计



## 四、文件结构

文件结构留言板的实现，主要利用MVC架构。使用以下命令完成数据库的创建python manage.py init\_database

1.数据库 model.py存储用户信息(用户名和密码)，存储用户的留言信息。

2. init.py初始配置

3. manage.py脚本启动

4.程序启动 run.py

5.登录注册 relogin；登录注册的init.py；登录注册的表单 form.py；

登录注册的 views.py；注册新用户时，密码需要加密，使用md5+salt的方式对密码加密。登录注册时，都会提交表单form，当form提交验证正确时，会进行登录注册下一步功能。主页上能够看到历史留言，分页显示；注册登录后可以留言，可以根据留言内容进行模糊查询，登录注册后可以在个人中心查看自己的留言，并且可以对自己的留言进行删除和修改。

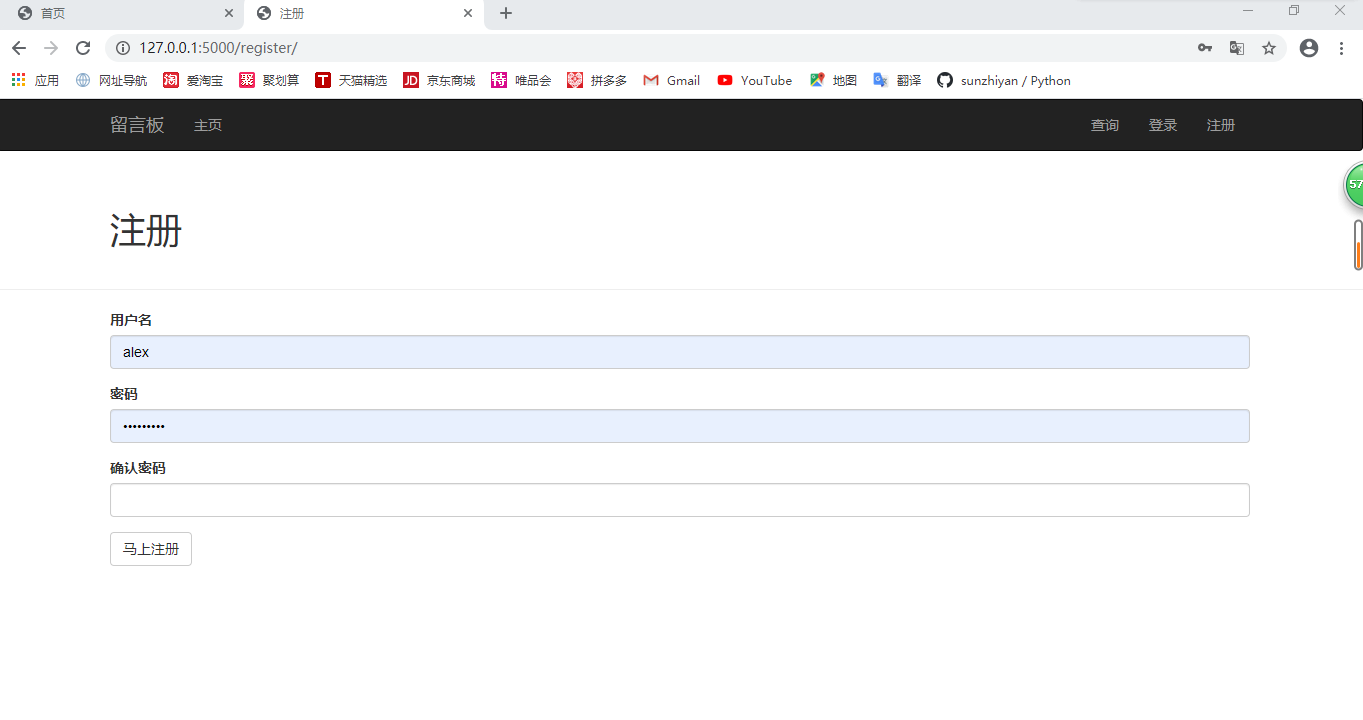
1. 留言 bbs；bbs的init.py；bbs的表单form.py；bbs的views.py；这里主要实现了主页的功能，主页上能够看到历史留言，登陆后可以留言
2. templates 模板本项目的留言板界面，主要继承bootstrap/base.html模板base.html；index.html，index为主页，主页上有分页显示的留言。

## 四、功能详细介绍

#### 1.注册

密码加密存储，将用户名和密码保存到user表中。代码如下

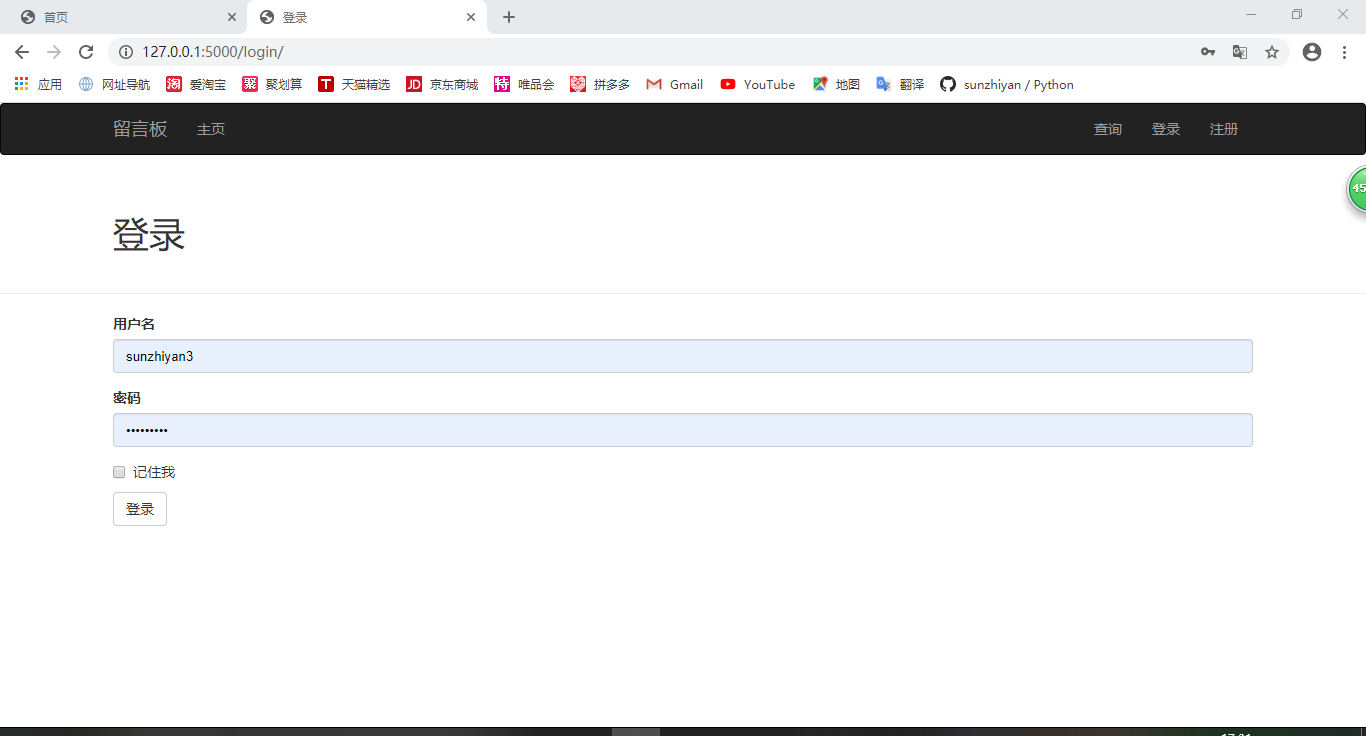
# 密码加md5+盐  
salt = ''.join(random.sample('0123456789abcdefghigklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ', 10))  
m = hashlib.md5()  
m.update((form.password.data + salt).encode("utf-8"))  
password = m.hexdigest()  
# 新用户提交入库  
user = User(form.username.data, password, salt)  
# 数据库提交  
db.session.add(user)  
db.session.commit()



#### 登录

验证密码的正确性，密码正确后登录，错误后提示密码错误，代码如下：

# 密码正确验证  
m = hashlib.md5()  
m.update((form.password.data + user.salt).encode("utf-8"))  
print  
if m.hexdigest() != user.password:  
 flash('密码输入错误')  
 return redirect('/login')  
login\_user(user)

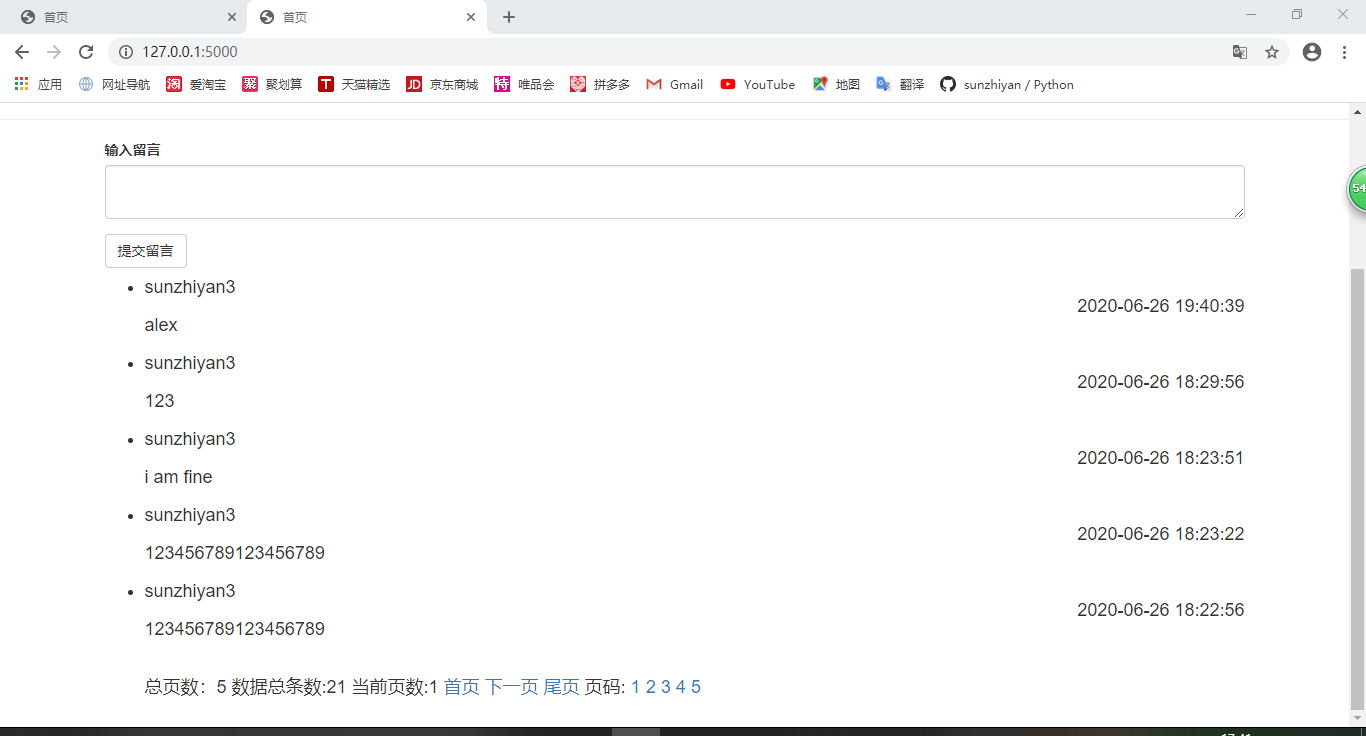


#### 留言

登陆后可以在主页留言，主页分页显示所有的留言信息。使用paginate分页。

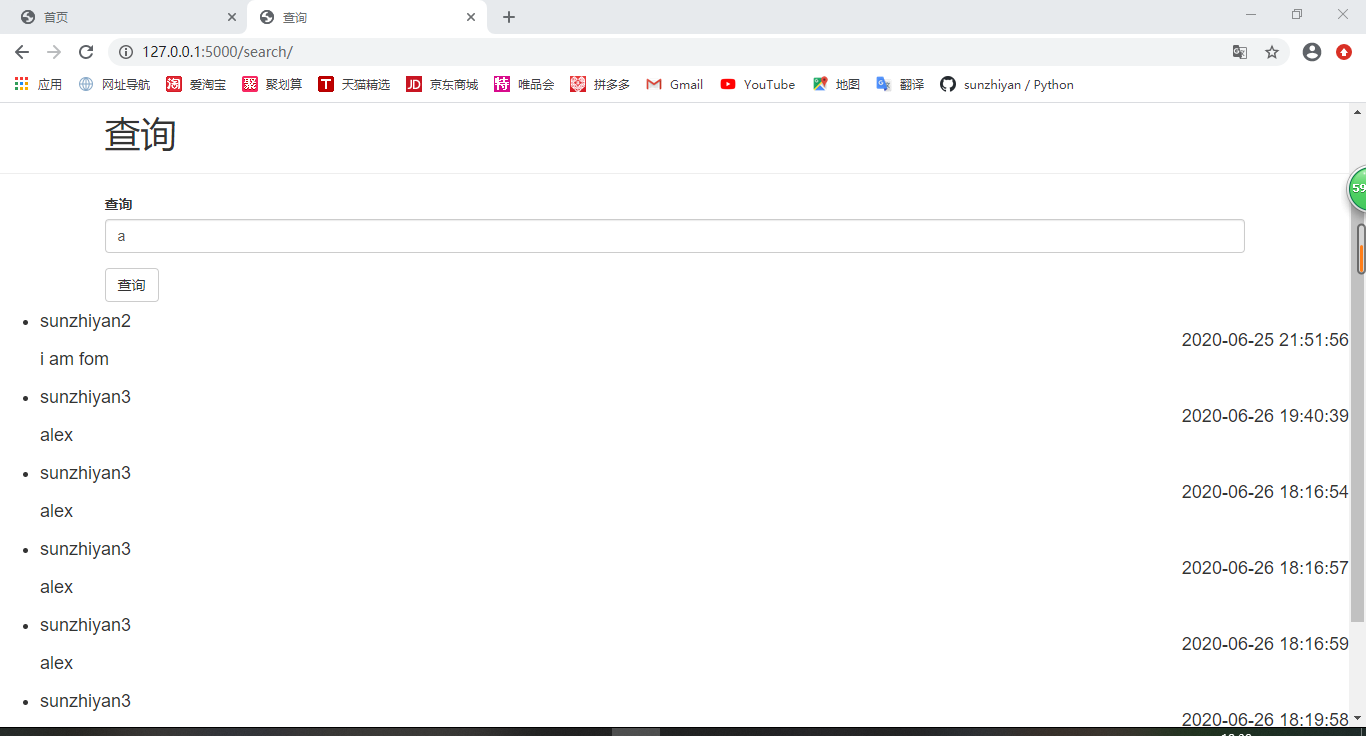
page = int(request.args.get('page', 1)) # 当前页数  
per\_page = int(request.args.get('per\_page', 5)) # 设置每页数量  
paginate = Message.query.order\_by(Message.timestamp.desc()).paginate(page, per\_page, error\_out=False)  
return render\_template('index.html', form=form, paginate=paginate)

<nav aria-label="Page navigation">  
 <ul class="pagination">  
  
 总页数：{{ paginate.pages }}  
 数据总条数:{{ paginate.total }}  
 当前页数:{{ paginate.page }}  
 <a href="/?page=1">首页</a>  
 {% if paginate.has\_prev %}  
 <a href="/?page={{ paginate.prev\_num }}">上一页</a>  
 {% endif %}  
 {% if paginate.has\_next %}  
 <a href="/?page={{ paginate.next\_num }}">下一页</a>  
 {% endif %}  
 <a href="/?page={{paginate.pages}}">尾页</a>  
 页码:  
 {% for i in paginate.iter\_pages() %}  
 <a href="/?page={{ i }}">{{ i }}</a>  
 {% endfor %}  
 </ul>  
</nav>



#### 查询

使用messages = Message.query.filter(Message.msg.like("%" + form.username.data + "%") if form.username.data is not None else "").all()对留言内容进行模糊查询



#### 个人中心

分页显示本人的所有留言，并且可以对留言进行删除和修改并保存到数据库中。代码如下：

message = Message.query.get(message\_id)  
try:  
 db.session.delete(message)  
 db.session.commit()  
except Exception as e:  
 print(e)  
 flash("删除成功")  
 db.session.rollback()

