

保密：绝密

文件编号：



# 动漫论坛系统 概要设计报告

版本：0.0.0-1.0.2

2020-10-22

昆山杰普软件科技有限公司 Web 与云计算部门

（版权所有，翻版必究）

保密：绝密

文件编号：



# 动漫论坛系统 概要设计报告

Web 与云计算部门

昆山杰普软件科技有限公司 Web 与云计算部门

（版权所有，翻版必究）

总页数		正文		附录		生效日期	
编制				批准			

修改记录

版本号	变更控制报告编号	更改条款及内容	更改人	审批人	更改日期
0.0.0-1.0.0		项目首次创建			
0.0.0-1.0.1		补全流程图和页面设计图			
0.0.0-1.0.2		术语表补充			

## 目录

<b>1 系统简介</b>	5
1.1 研究的背景	5
1.2 研究现状及发展趋势	5
1.3 系统环境需求	5
<b>2 开发环境及相关技术介绍</b>	5
2.1 Django 框架介绍	6
2.2 Python 介绍	6
2.3 REST 规则	6
2.4 MySQL 介绍	6
2.5 前端技术介绍	6
<b>3 功能介绍</b>	7
3.1 系统基本功能介绍	7
3.2 面向游客用户的功能模块	7
3.3 面向管理员的功能模块	8
3.4 系统总体功能模块设计	8
<b>4 系统设计与实现</b>	9
4.1 系统总体需求	9
4.2 系统总体功能模块设计	9
4.3 数据库设计说明	10
<b>5 系统演示</b>	12
5.1 登录注册功能	12
5.2 用户管理页面功能	15
5.3 首页页面功能	17

## 1 系统简介

### 1.1 研究的背景

近几年来,随着网络在全球的发展与应用,动漫论坛的发展非常的迅速。尤其是图形、声音、动画等技术的不断完善与发展,我们不仅可以看到简单的文字和图片,还可以有根据的找到自己感兴趣的动漫论坛,发表自己的意见或者建议,增加论坛的互动性和丰富性。

本系统设计方案包括动漫论坛的设计与实现,以及介绍本系统所运用到的相关技术、开发环境,数据库设计,对各个功能模块的需求分析和详细设计,系统角色大致分为游客,用户,管理员。系统功能模块大致分为:登录注册模块,首页模块,用户管理模块,帖子管理模块,日志管理模块,个人中心管理模块。

通过对这些功能模块的实现与应用,不仅可以强化我在大学和培训基地中所学的知识,并且可以对软件开发的思想和软件开发的过程有一个详细深入的认识,使我可以在此系统的设计过程当中得到很强的锻炼。

### 1.2 研究现状及发展趋势

当今的网络化时代,在互联网上轻轻一点,每个人几乎都可以找到自己喜欢或者需要了解的专题性论坛,而各类网站,综合性门户网站或者功能性专题网站也都青睐于开设自己的论坛,以促进网友之间的交流,增加互动性和丰富网站的内容。动漫作为文化产业的一部分,他影响了很多,包括各个年龄段的人群。动漫论坛是以动漫交流为主题的 BBS 网上虚拟社区,他是动漫爱好者们共享信息,互相交流的平台。而这种论坛服务,迄今已经历了三十多年,已经成为人们网络生活中必不可少的一部分,并且越来越多元化,产生了很多其它的方式,如百度贴吧,知乎等。

动漫论坛系统,可以使有相同兴趣爱好的人,聚集在一起,打破了时间、空间的界限,网友们可以互相交流自己感兴趣的动漫,并可以简单便捷的观赏自己喜欢的动漫,从基本上促进了动漫的发展和传播。

### 1.3 系统环境需求

#### (1) 硬件环境

处理器: Intel(R) Core(TM)i5-6200U CPU @2.30GHz 2.40GHz

内存: 4.00 GB 以上

#### (2) 软件环境配置

操作系统: Windows10 (推荐) / Windows7

数据库: MySQL

开发环境: Python3.7, Django2.0

开发语言: Python、JavaScript

浏览器: 火狐 (推荐) / 谷歌/IE 浏览器

## 2 开发环境及相关技术介绍

本系统全称为基于 Django 的动漫论坛的设计与实现,开发的目的是为了用户的闲置时间得以欣赏最新的动漫信息以及每个动漫迷的思想观点,可充分反应最广大人民群众的看法,在一定程度上给予用户最好的感受。

本文所采用的技术方案是基于 B/S 架构,开源的后台框架 Django,开发语言使用的是 Python 语言,数据库使用的是开源免费的关系型数据库 MySQL,前端界面显示使用 HTML、CSS、JavaScript、Ajax 等前端语言技术。由于以上技术都是开源,免费,并且拥有活跃的社区和十分广阔的资源库,有利于相关资料的查询和收集,以及没接触过的技术点可以进行查找讨论。以下主要介绍本系统开发中所用的主要技术。

## 2.1 Django 框架介绍

Django 是面向对象的解释性脚本语言 Python 开发的开源 Web 框架，它继承了很多有用的库，还有相关的开发人员根据自己的需要而开发的第三方插件，并且拥有非常成熟的 URL 设计，避免新的开发中根据共性问题开发应用，使开发者可以把精力放在程序核心功能的开发，编写出通俗易懂，有利于维护的代码。

在 Django 框架中模板是相互独立的，支持许多种文档数据类型，不仅仅可以输出 HTML 页面，并且模板代码也可以重复使用。但是由于本文采用 REST 设计规范，所以并不适用 Django 内置的模板来进行网页的显示。

在 Django 框架中模型的设计采用 Python 代码来定义数据格式，通过 ORM 技术关联数据库表，并且 ORM 技术是可以将 Django 中编写好的数据模型自动映射到数据库类型中，并将建立起表与表之间的关系（外键关联、一对一、多对多），并与 MySQL 相同的字段类型匹配。我们可以调用 Django 模型中的一些内置方法来操作对应数据库中的记录，减少 MySQL 语句的编写。

在 Django 框架中视图主要做的事是根据系统的业务逻辑从数据库中拿到数据并且对数据进行处理，然后将这些数据传递给前端页面当中。

Django 后台管理，Django 可以产生一个超级管理员用户对后台全部数据进行增删改查的操作<sup>[3]</sup>。

## 2.2 Python 介绍

Python 是一种简单的，解释型，交互式的，可移植的，面向对象的超高级语言，Python 作为一种功能强大且通用的编程语言而广受好评，它具有非常清晰的语法特点，使用与多种操作系统。常用数据类型：字符串、模块、函数、数字都是可操作对象，也可以和 Java 一样支持类的继承，实例化极大提高了代码的扩展性和重复利用率，方便了开发者使用最简单最少的代码开发用户所需的功能。目前在国际上非常流行，正在得到越来越多的应用。

Python 拥有十分完善的基础应用库，拥有数据库编程操作、多线程编程等。使用 Python 进行程序开发的时候，大部分功能都不需要从新编写，直接调用已存在的成熟的库即可。Python 经过 30 年的发展，许多第三方公司和开发者为 Python 编写了接口并封装成 Python 扩展库，供所有开发者使用<sup>[5]</sup>。

## 2.3 REST 规则

REST 含义为表述性状态转移（Representational State Transfer）是一种软件架构风格。通过 URL 对所有资源进行唯一标识，通过前端的 Ajax 技术轻松连接。定位和使用资源。

通过 REST 规则可以用来开发前后端分离的系统，后端仅返回前端需要的数据，不需要渲染 HTML 页面，不再控制前端的效果。从后端请求的数据如何加载到前端中，都由前端自行决定。这样网站、APP 等前端所需的数据都可以通过一套 API 接口来提供，前端通过访问这个接口来对数据进行增删改查。在前后端分离的模式中，这种规则使得前后端更专注于编写自己的代码，使系统的完成效率提高。

Django Rest Framework 就是基于 Django 和 Python 应用 REST 规则开发的一个强大灵活的工具包，用以构建 Web API。

## 2.4 MySQL 介绍

MySQL 是瑞典的 MySQL AB 公司开发的产品。他是一个跨平台的，开源的小型关系型数据库管理系统，可用于各种流行操作系统平台的关系数据库系统。MySQL 在过去凭借着性能高、成本低、可靠性好的原因，成为了最流行的开源数据库，所以被广泛地应用在 Internet 上的中小型网站中。后来随着 MySQL 的不断成熟与发展，它也逐渐用于更多大规模网站和应用中。

## 2.5 前端技术介绍

前端页面主要采用了 Layui 框架，其中代码的编写运用到了 HTML、CSS、JavaScript、AJAX 等方面的语言和技术来进行设计和实现。以下主要介绍前端中所用到的主要技术。

1). Layui: 是一个前端集成度很高和功能完善的框架，符合基础 HTML/CSS/JS 的编写规范，使用起来特别简单，适合界面的快速开发除此之外，事实上，他更多面向于后端开发者，因此，在组织形式

上采用了以浏览器为宿主的类 AMD 模块管理方式，却并非受限于那些条条框框，它拥有自己的模式，更加简单

2). HTML:它是一种描述网页的语言，是超文本标记语言，缩写为 HTML,他是网页制作所必备的工具，用来编写网页，描述文本、图片、视频的内容，主要组成部分有根标记、头部标记和主体标记，它通过标记符号来显示网页中需要标记的部分。

3). CSS:它是层叠样式表，英文是 Cascading Style Sheet,它是用来修饰 HTML 或 XML 的一种计算机语言。它的作用是增强网页的显示效果，使 HTML 代码及显示出来的页面清晰有条理。拥有丰富的样式定义，也非常便于使用和修改，可以多页面应用。使用 CSS，让网页内容和样式分离，便于网站代码后期的维护。（CSS 不能单独使用，必须和 HTML 网页配合使用）

4). JavaScript:诞生于 1995 年，是一种动态直译，弱类型，基于对象和事件驱件的语言，广泛应用于浏览器，客户端等场景，它是来来给 HTML 页面增加动态效果。

5). AJAX:AJAX 其实是基于 JavaScript 的一种技术来实现页面的刷新，使用户在体验页面刷新时更平滑。

3 功能介绍

3.1 系统基本功能介绍

系统设计中必须要实现的具体功能以使用户和管理员的使用和操作,这些功能要满足当初根据甲方客户提出来的系统最基本的需求，将系统做成用户所期望的样子，本小节主要列出一些要实现的功能模块。

系统的基本模块如表 3.1 所示：

功能名称	描述
首页	用户在系统上的首页对于本身的一些操作以及对于自身帖子和他人帖子的一些操作，查看和维护，管理员对于一些用户帖子的查看以及对于系统的维护。
用户管理	管理员对于注册过的用户进行一些基本的操作。
帖子管理	管理员对于注册过的用户所发表的帖子进行一些基本的操作和维护。
日志管理	管理员及用户实现对自身进行过的操作内容进行查看和维护，以及管理员对于系统注册用户的历史进行查看和维护。

表 3.1 系统基本功能

3.2 面向游客用户的功能模块

- (1)注册登录：可以在前台注册（游客注册成为用户）和登录。
- (2)帖子查询与浏览：可以在首页进行帖子查询以及对系统上的所有帖子进行浏览。
- (3)自身帖子管理：可以在首页进行登录后发表自己的帖子，以及对自己的帖子进行一定的操作和维护。
- (4)帖子评论与回复：可以在用户发表的帖子以及首页上对其它用户发表的帖子进行评论，也可以在其他用户的评论下进行评论。
- (5)个人信息管理：对用户本身信息的一些操作。

### 3.3 面向管理员的功能模块

- (1) 首页管理：包括对用户发表帖子的信息的一些操作。
- (2) 用户管理：用户管理包括对用户的一些操作。
- (3) 帖子管理：帖子管理包括用户发表的帖子及与该帖子相关的评论和回复的删改查。
- (4) 日志管理：日志管理记录了系统用户注册的记录及管理员对于系统的操作。

### 3.4 系统总体功能模块设计

基于 Django 的动漫论坛的设计与实现主要由登录，注册，用户管理，帖子管理，日志管理，首页，个人信息管理几个功能模块组成。

本系统各个模块的功能详情如表 3.2 所示：

模块	功能
登陆管理	验证用户登录
注册管理	用于游客注册成为用户
用户管理	查询用户信息
	修改用户信息
	冻结用户
	删除用户
帖子管理	查询帖子信息
	修改帖子信息
	冻结帖子
	删除帖子
日志管理	添加日志
	修改日志
	删除日志
	查询日志
首页	登录用户浏览他人帖子
	登录用户评论他人帖子
	登录用户回复他人帖子
	登录用户搜索他人帖子
	登录用户查看他人信息
	登录用户自身帖子的增删改查
	登录用户自身信息的查看修改
	登录用户自身的注销
个人信息管理	用户自身信息的查看与修改
	用户自身的注销

表 3.2 系统总体模块功能表

登录模块用于验证用户和管理员的登录并根据身份展示不同的页面，注册模块用于游客的基本信息填写，以便于转化为用户身份从而使用本系统的功能，用户管理模块用于管理员对系统内的用户进行一定的操作和维护，帖子管理模块用于管理员对系统内的用户发表的帖子进行一系列的操作和维护，日志管理是对系统的有关操作和维护的一个记录方便后续管理员的操作和维护，首页是展现给用户的页面，主要是面对用户的使用和用户对于自身的一些操作以及对其他用户信息的基本浏览，个人信息管理是用户对自己本身的一系列的操作和维护。



4 系统设计与实现

4.1 系统总体需求

本系统总体需求的用例图如图 4.1 所示：

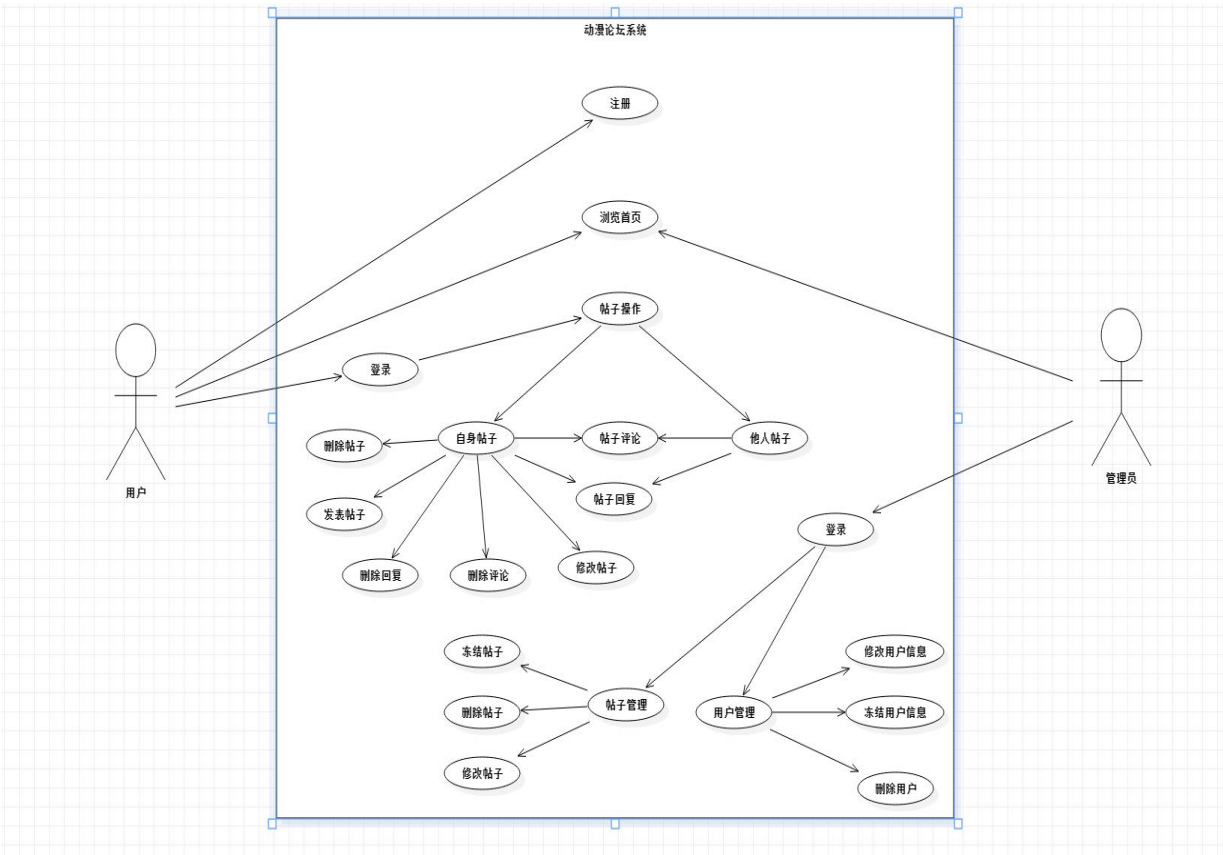


图 4.1 系统总体需求用例图

4.2 系统总体功能模块设计

基于 Django 的动漫论坛的设计与实现主要由登录，注册，用户管理，帖子管理，日志管理，首页，个人信息管理几个功能模块组成。

本系统各个模块的功能详情如表 4.1 所示：

模块	功能
登陆管理	验证用户登录
注册管理	用于游客注册成为用户
用户管理	查询用户信息
	修改用户信息
	冻结用户
	删除用户
帖子管理	查询帖子信息
	修改帖子信息
	冻结帖子
	删除帖子
日志管理	添加日志

	修改日志
	删除日志
	查询日志
首页	登录用户浏览他人帖子
	登录用户评论他人帖子
	登录用户回复他人帖子
	登录用户搜索他人帖子
	登录用户查看他人信息
	登录用户自身帖子的增删改查
	登录用户自身信息的查看修改
	登录用户自身的注销
个人信息管理	用户自身信息的查看与修改
	用户自身的注销

表 4.1 系统总体模块功能表

登录模块用于验证用户和管理员的登录并根据身份展示不同的页面,注册模块用于游客的基本信息填写,以便于转化为用户身份从而使用本系统的功能,用户管理模块用于管理员对系统内的用户进行一定的操作和维护,帖子管理模块用于管理员对系统内的用户发表的帖子进行一系列的操作和维护,日志管理是对系统的有关操作和维护的一个记录方便后续管理员的操作和维护,首页是展现给用户的页面,主要是面对用户的使用和用户对于自身的一些操作以及对其他用户信息的基本浏览,个人信息管理是用户对自己本身的一系列的操作和维护。

#### 4.3 数据库设计说明

名字: 用户表 描述: 用户的具体详情信息 定义: 用户信息 = 编号+用户名+登录密码+用户手机号+用户邮箱+用户性别+用户状态+用户角色+添加时间+最后登录时间 编号: 数据库自动产生 用户名: 多个汉字 登录密码: 多个数字, 英文字母, 标点符号 用户手机号: 多个数字 用户邮箱: 标准格式, 例如: 1145@qq.com 用户性别: 男或女 用户状态: 活跃或冻结 用户角色: 用户或管理员 添加时间: datetime 格式, 必须是当前时间前的日期 例如: 2020-05-25 02:15:47.814105 最后登录时间: datetime 格式
名字: 帖子信息表 描述: 帖子的具体详情信息 定义: 帖子信息 = 编号+帖子名称+帖子说明+帖子内容+添加时间+发表人 编号: 数据库自动产生 帖子名称: 多个汉字 帖子说明: 多个汉字 帖子内容: 多个汉字

添加时间: datetime 格式, 必须是当前时间前的日期 例如: 2020-05-25 02:15:47.814105 发表人: 关联用户表中的用户
名字: 帖子评论信息表 描述: 帖子评论的具体详情信息 定义: 帖子评论信息 = 编号+评论内容+评论时间+评论人+所属帖子 编号: 数据库自动产生 评论内容: 多个汉字 评论时间: datetime 格式, 必须是当前时间前的日期 例如: 2020-05-25 02:15:47.814105 评论人: 关联用户表中的用户 所属帖子: 关联帖子信息表中的帖子
名字: 帖子回复信息表 描述: 帖子回复的具体详情信息 定义: 帖子回复信息 = 编号+回复内容+回复时间+回复人+所属评论 编号: 数据库自动产生 回复内容: 多个汉字 回复时间: datetime 格式, 必须是当前时间前的日期 例如: 2020-05-25 02:15:47.814105 回复人: 关联用户表中的用户 所属评论: 关联帖子评论信息表中的评论
名字: 日志表 描述: 日志信息的具体详情信息 定义: 日志信息=编号+添加时间+日志内容+操作用户 编号: 系统自动生成 添加时间: datetime 格式, 必须是当前时间前的日期 例如: 2020-05-25 02:15:47.814105 日志内容: 字符串 用户: 关联用户表中的用户

表 4.2 数据库设计说明表

根据以上设计以及实际开发情况, 给出既定数据库表及字段定义如下:

(1) **users\_usersmodels**: 用户表, 如表 4.3 所示, 用户信息: id, 密码, 用户名, 用户状态, 邮箱, 电话, 性别, 角色。

字段名	数据类型	长度	可否为空	键引用	字段含义	备注
id	int	11	否	主键	ID	
password	varchar	128	否			
username	varchar	150	否			
user_status	int	6	否			
email	varchar	6	否			
user_tel	varchar	30	否			
user_gender	bool	6	否			
user_role	int	255	否			

表 4.3 用户表

(2) **invitations\_invitation**: 帖子信息表, 如表 4.4 所示, 帖子信息: id, 帖子名称, 帖子说明, 帖子内容, 添加时间, 添加用户。

字段名	数据类型	长度	可否为空	键引用	字段含义	备注
-----	------	----	------	-----	------	----

动漫论坛系统

id	int	11	否	主键
inv_name	varchar	6	否	
inv_detail	varchar	6	否	
inv_text	varchar	30	否	
inv_addtime	datetime	30	否	
user	fk	10	否	外键

表 4.4 帖子信息表

(3) invitations\_discuss: 帖子评论信息表, 如表 4.5 所示, 帖子评论信息: id, 评论内容, 评论时间, 所属帖子, 评论用户。

字段名	数据类型	长度	可否为空	键引用	字段含义	备注
id	Int	11	否	主键		
dis_text	Varchar	30	否			
dis_addtime	Datetime	30	否			
dis_inv	fk	10		外键		
dis_user	fk	10	否	外键		

表 4.5 帖子评论信息表

(4) invitations\_reply: 帖子回复表, 如表 4.6 所示, 帖子回复信息: id, 回复内容, 回复时间, 所属评论, 回复用户。

字段名	数据类型	长度	可否为空	键引用	字段含义	备注
id	int	11	否	主键		
re_text	varchar	30	否			
re_addtime	datetime	30	否			
re_dis	fk	10	否	外键		
re_user	fk	10	否	外键		

表 4.6 帖子评论信息表

(5) logs\_log: 日志表, 如表 4.7 所示, 日志信息: id、添加时间、日志详情、添加用户。

字段名	数据类型	长度	可否为空	键引用	字段含义	备注
id	int	11	否	主键		
log_addtime	datetime	255	否			
log_context	varchar	255	否			
user	fk	1	否			

表 4.7 日志表

5 系统演示

5.1 登录注册功能

登录注册模块是基于 Django 的动漫论坛管理系统的入口和身份的申请, 如果用户没有使用账号或者用户名、密码登录或者密码错误, 都将不能为访问动漫论坛管理系统, 填写的信息会有格式要求。登录界面如图 5.1 所示:

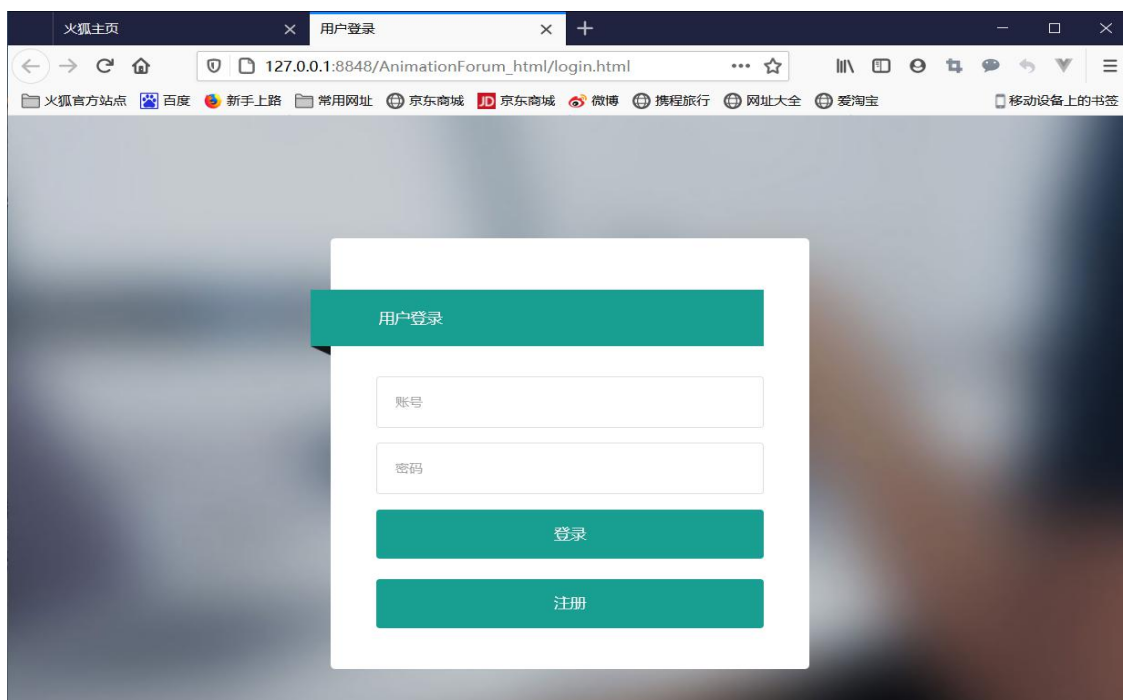


图 5.1 登录界面

如果没有注册过的游客就可以点击注册按钮进行注册,填写必要信息成功注册的用户会自动返回到登录界面,然后进行用户名和密码的统一验证,如果验证成功就会进入到系统内,注册界面如图 5.2 所示:

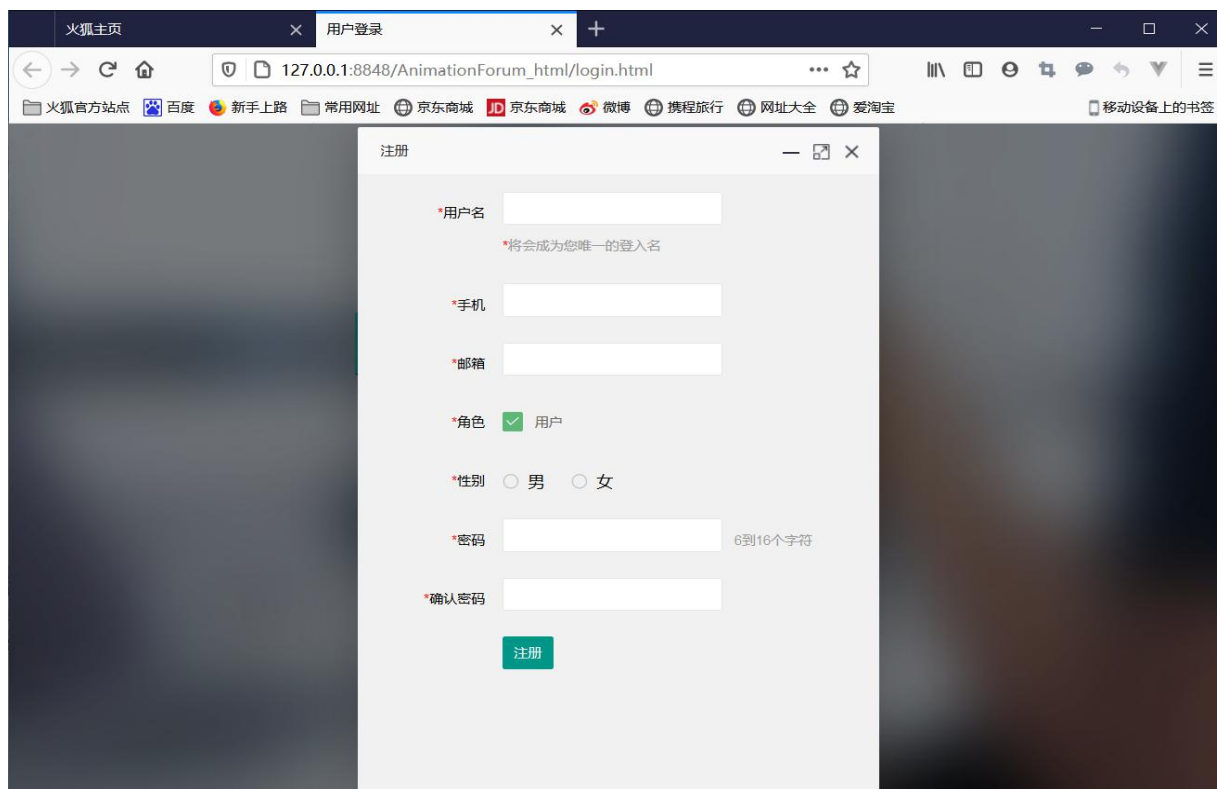


图 5.2 注册界面

登录界面中如果用户不输入密码或者输入错误,系统显示报错信息,用户需要重新填写相关信息。如图 5.3 所示

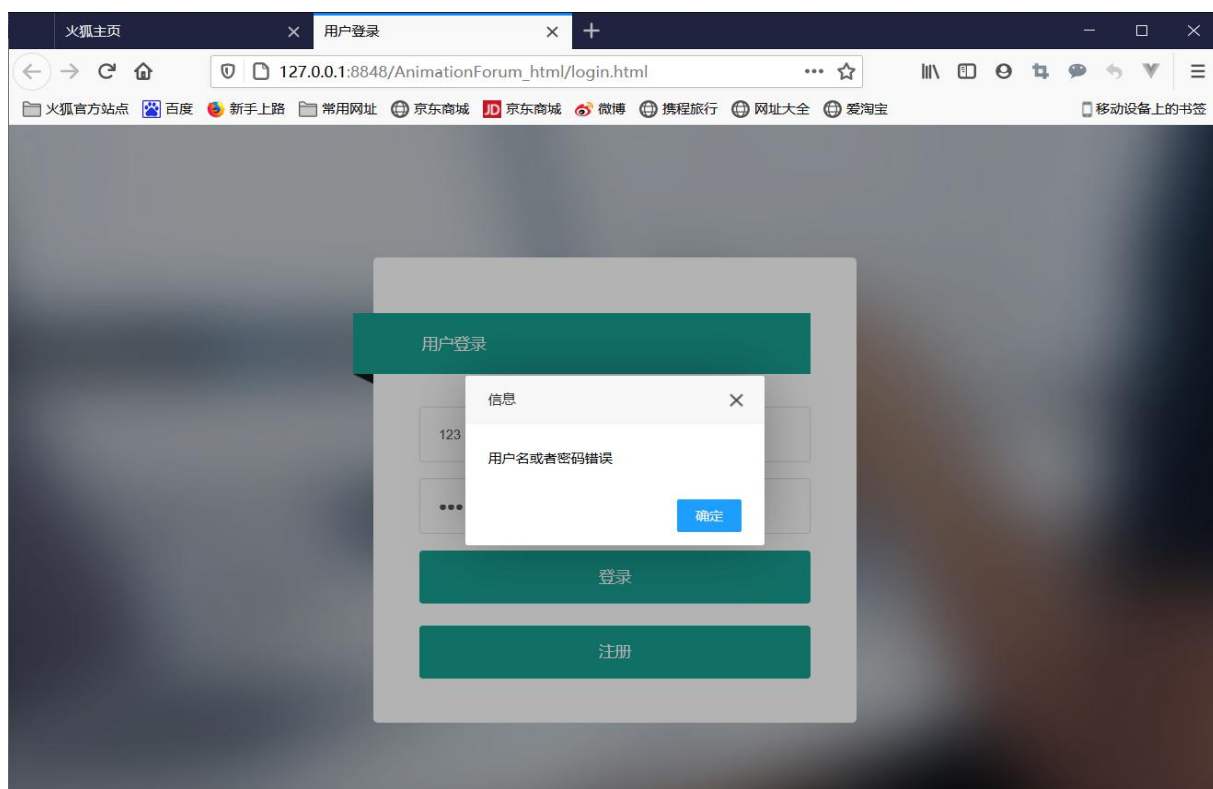


图 5.3 登录信息错误界面

系统对用户的密码采用的是默认加密算法：pbkdf2\_sha256，存储在数据库中，前台采用了对数据格式的验证，后台也进行用户验证及验证后的处理，这样既保证了用户的体验，又增加了系统的安全性。在用户输入账号密码登录系统时，系统会对用户和密码是否正确进行判断，分配给用户一个 7 天过期的 Token 头部信息作为用户的认证信息后携带该头部信息跳转到统计模块对应的页面中。具体流程如图 5.4：。

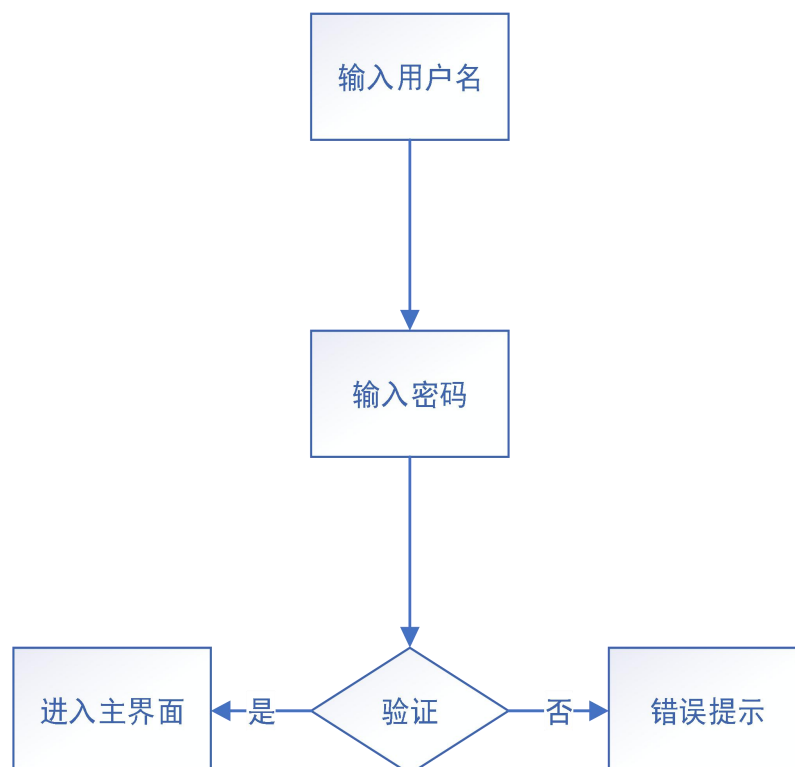


图 5.4 登录处理流程

登录模块程序的处理流程大概如下：1. 后台程序接收前台用户提交的登录信息，2. 系统与数据库中的登录信息进行匹配，3. 如果匹配成功，后台程序会为该用户发放一个 token 头部信息，并跳转到主页面中，如果验证失败，会返回错误信息到前台，提醒用户重新输入登录信息。

5.2 用户管理页面功能

本模块的功能是将用户的信息展示给管理员以便管理员一些进行用户信息的操作，管理员通过点击左侧导航栏的用户管理进入此模块，并通过右侧的操作按钮对注册过的用户进行一定的操作，并将结果上传到后台，后台在解析后将用户信息存入到数据库中。

用户的基本信息的展示以及用户管理页面的基本使用信息如图 5.5 所示：

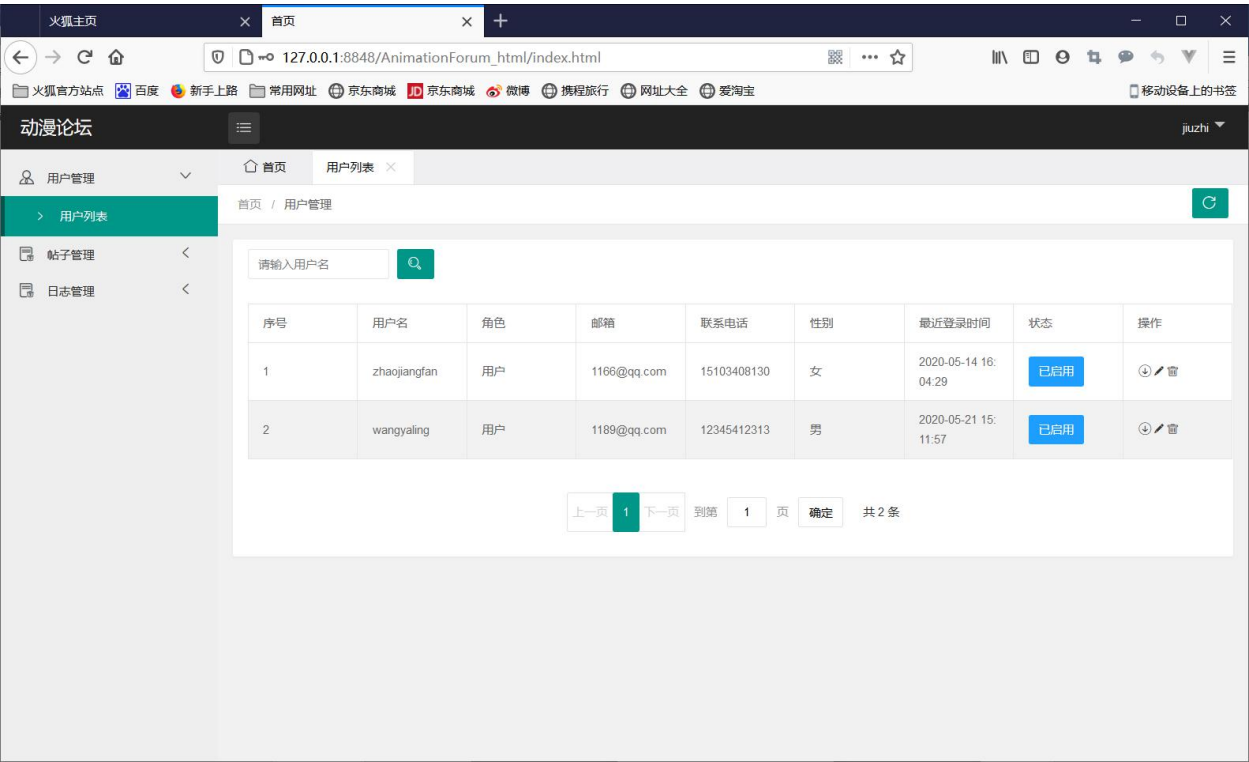


图 5.5 用户管理页面

管理员根据页面上的修改按钮修改用户信息，填入表单中的字段，都为必填并且格式需要正确。经过后台验证无误后便可修改数据库中的内容，并返回刷新后的当前页面。修改用户信息的页面如图 5.6 所示：

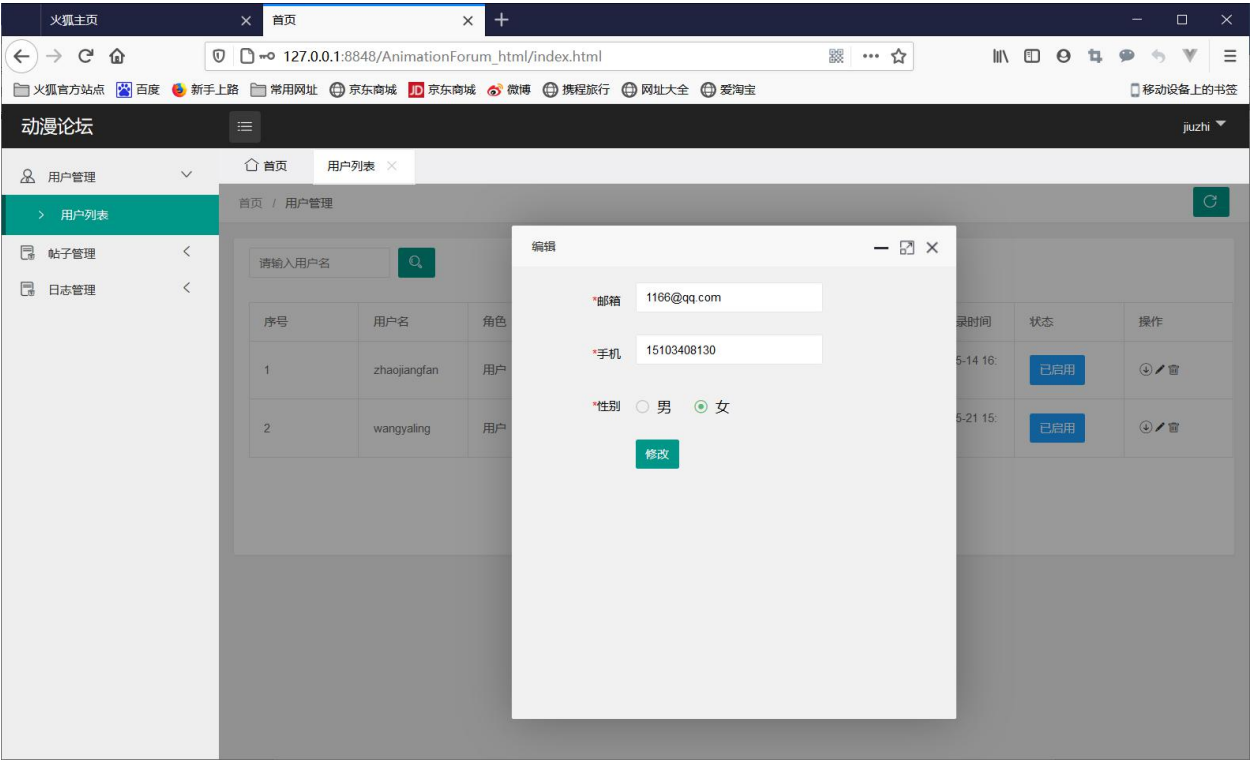


图 5.6 用户信息修改功能

管理员根据页面上的删除按钮删除用户信息，点击确认删除后经过后台验证无误后便可从数据库中删除该用户的信息，并返回刷新后的当前页面。删除用户信息的页面如图 5.7 所示：

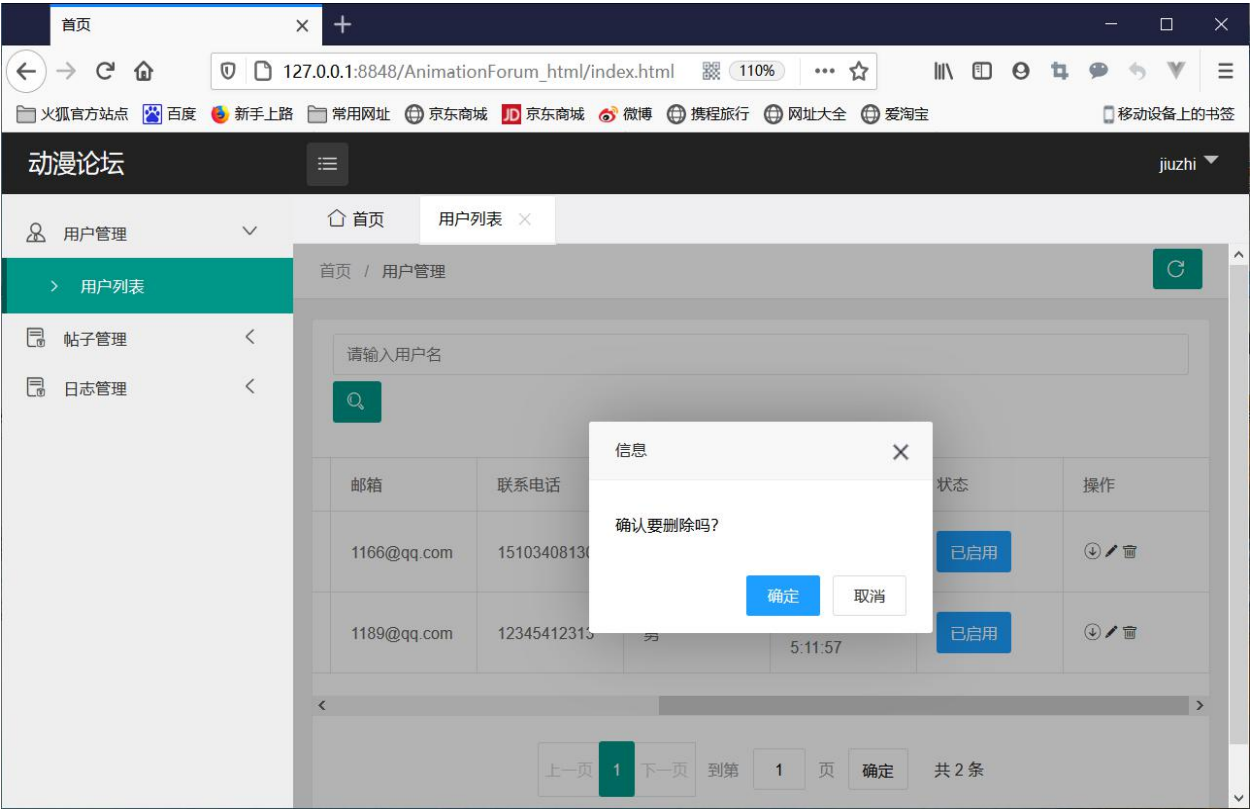


图 5.7 用户删除功能

管理员根据页面上的冻结按钮冻结用户，点击确认冻结后经过后台验证无误后便可从数据库中修改该用户的状态，并返回刷新后的当前页面，冻结用户信息的页面如图 4.8 所示：



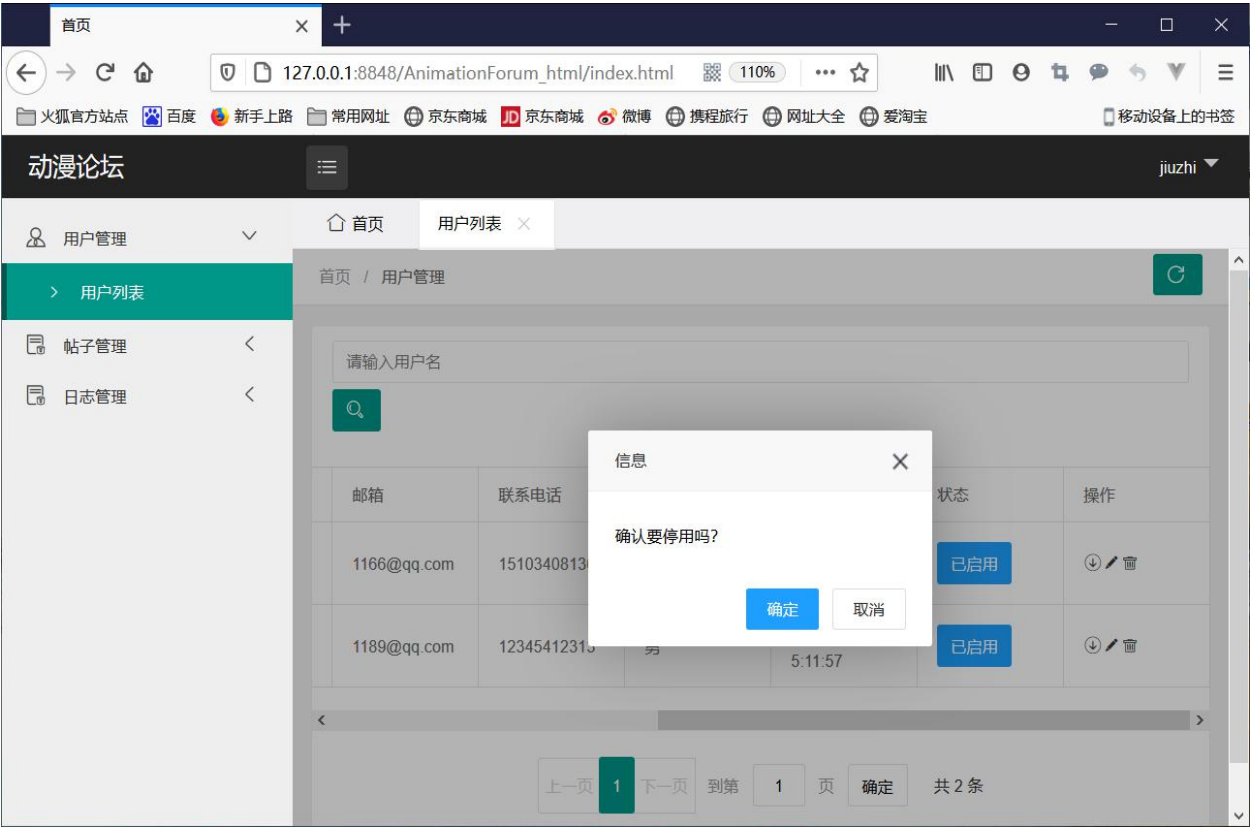


图 5.8 用户冻结功能

### 5.3 首页页面功能

本模块的功能是将首页上的信息展示给用户，用户在此页面可以查询某用户的信息，以便浏览该用户的信息以及该用户的帖子信息，用户在此页面可以大概查看个人的信息，并可以对自身的信息进行修改，以及对他人帖子进行评论回复及删除，用户通过对该系统的登录进入此模块，并通过页面上的操作按钮进行一定的操作，并将结果上传到后台，后台在解析后将信息存入到数据库中，并及时刷新网页。

点击首页展示，系统将首页基本信息展示给用户以及提供用户在此页面上的基本操作，具体信息如图 5.9 所示：

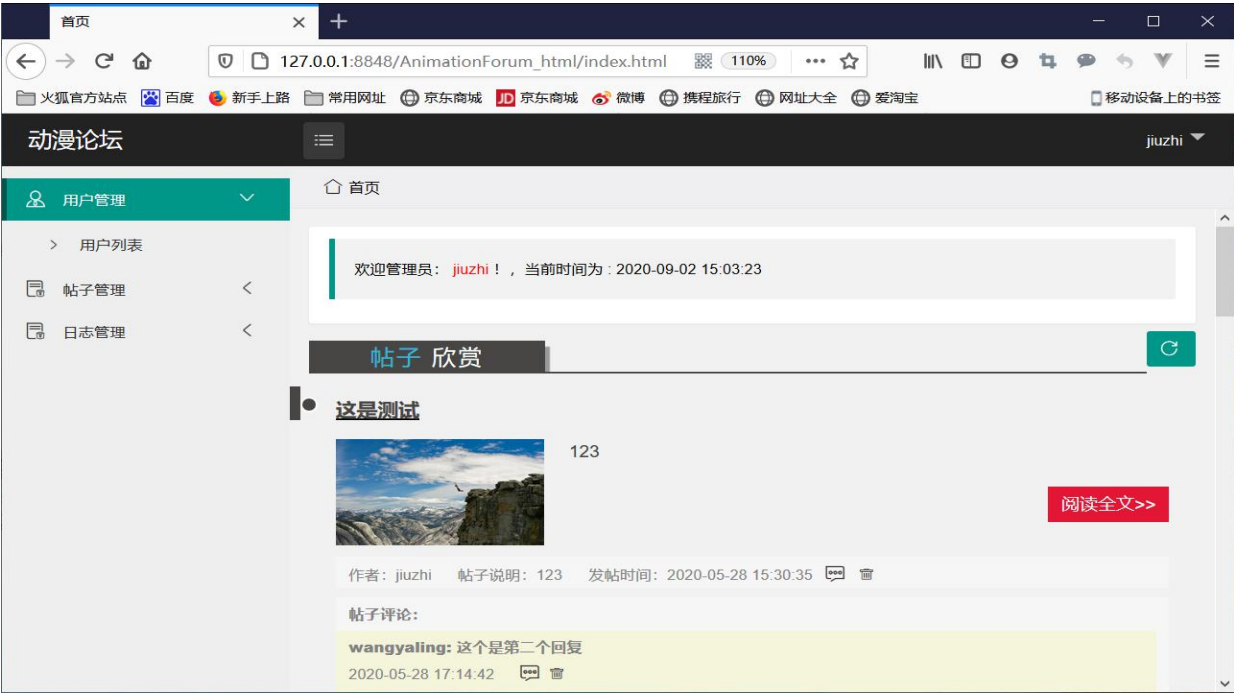


图 5.9 首页基本信息

用户在首页上点击帖子评论按钮后，系统弹出一个添加评论的悬浮框，用户在悬浮框上面的评论内容中进行编写评论的内容，然后点击确定按钮或者点击右上角退出评论按钮，具体内容如图 5.10 所示：

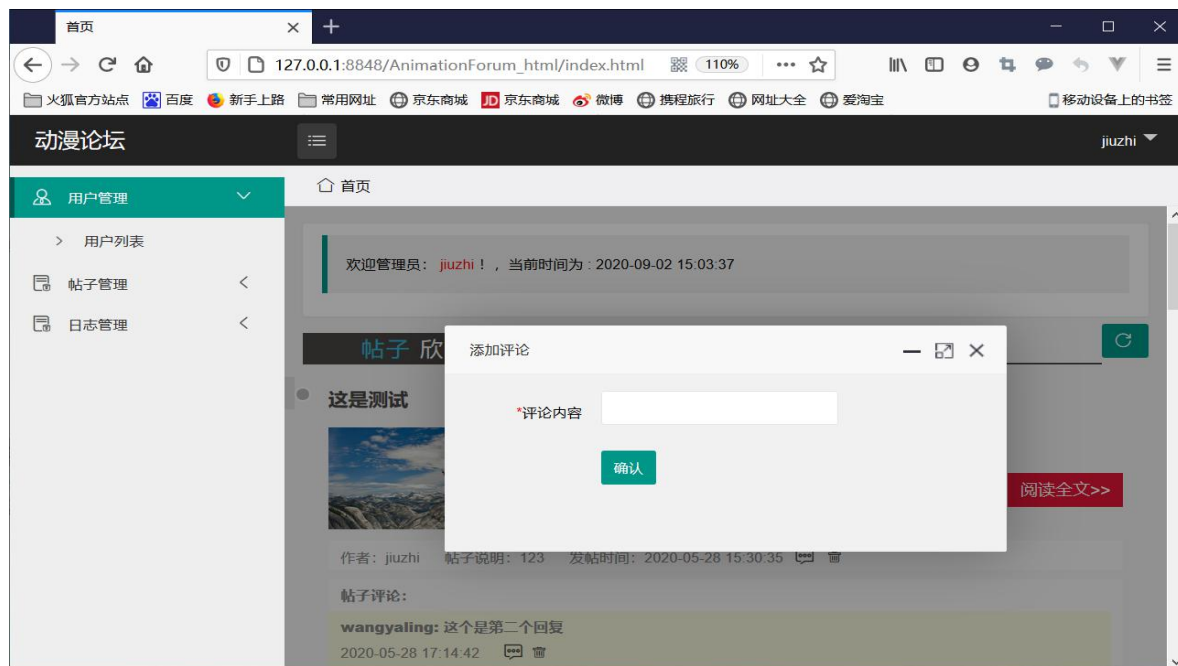


图 5.10 帖子评论

用户在首页上点击回复帖子评论按钮后，系统弹出一个添加回复评论的悬浮框，用户在悬浮框上面的回复内容中进行编写回复的内容，然后点击确定按钮或者点击右上角退出回复按钮，然后系统根据用户的操作重新刷新页面展示给用户，具体信息如图 5.11 所示：

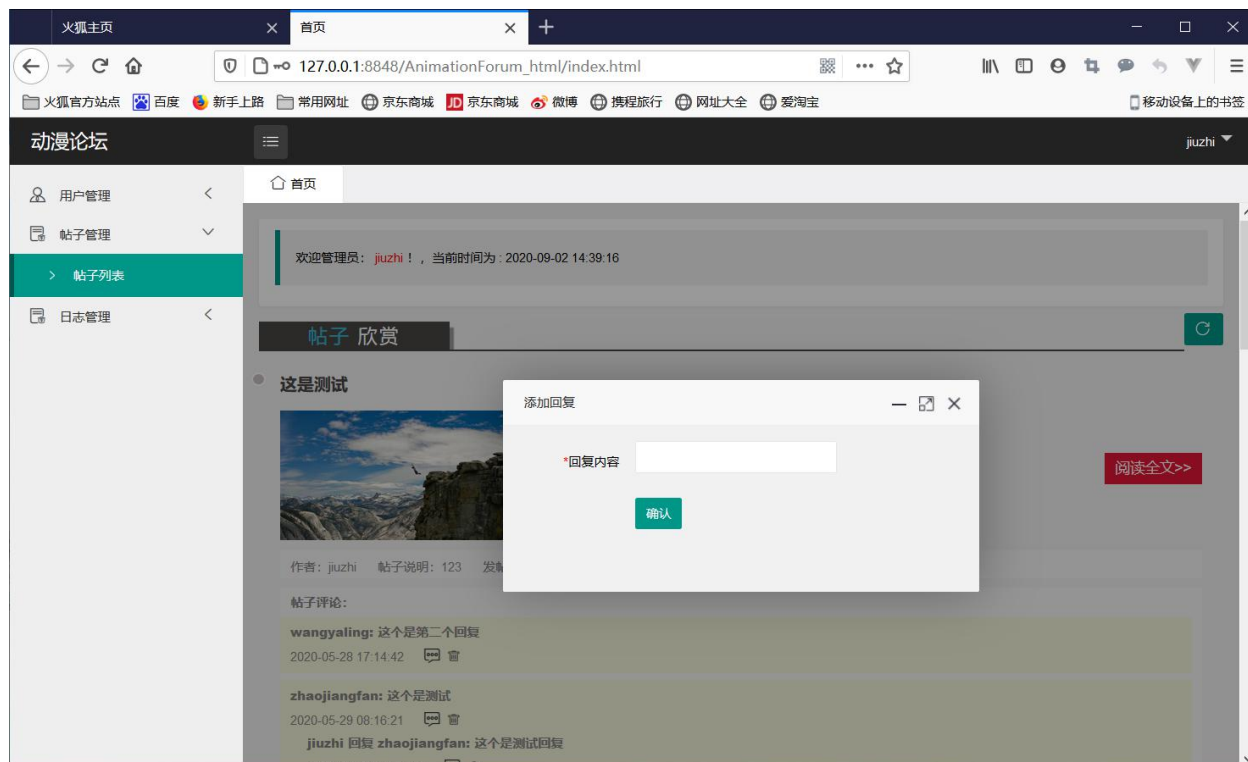


图 5.11 帖子回复

用户在首页上点击删除帖子按钮后，系统会弹出一个是否删除帖子的提醒框，用户在提醒框上面进行确认删除操作或者取消删除操作，具体帖子删除基本使用信息如图 5.12 所示：

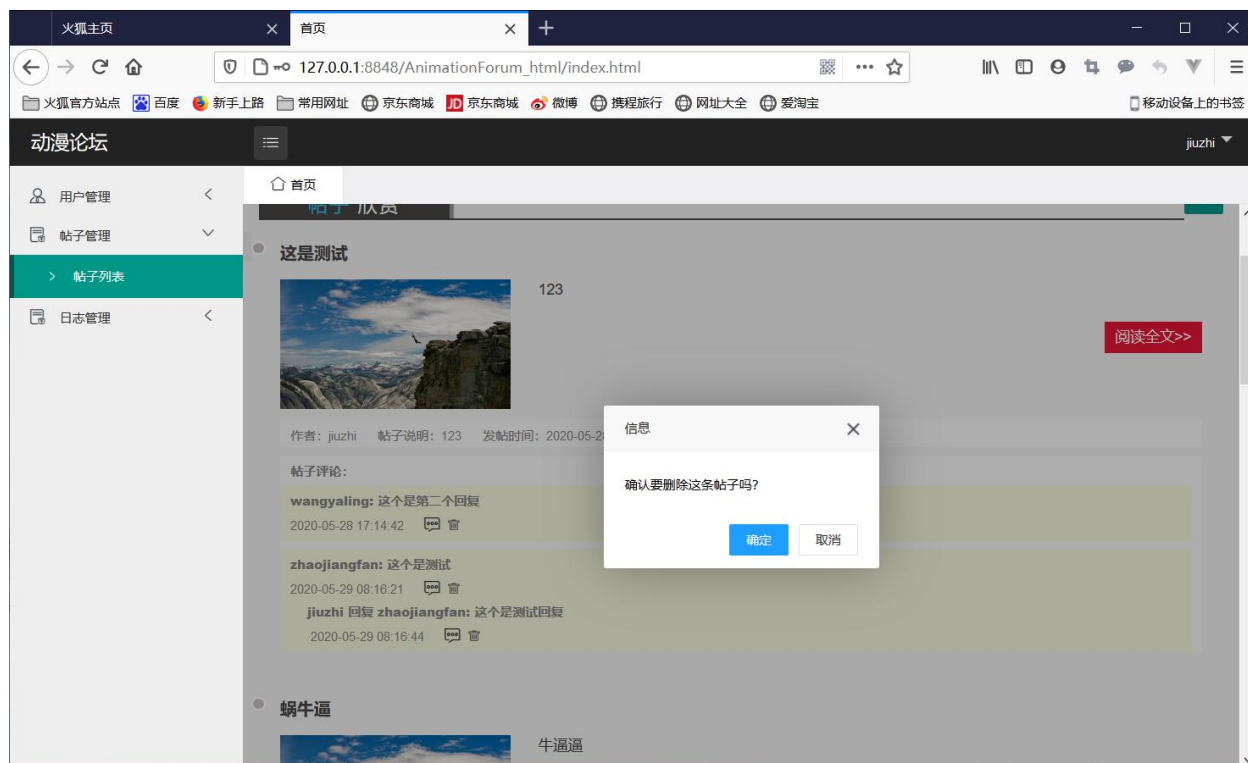


图 5.12 删除帖子

用户在首页上点击删除帖子评论按钮后，系统会弹出一个是否删除评论的提醒框，用户在提醒框上面进行确认删除操作或者取消删除操作，具体删除帖子评论的基本使用信息如图 5.13 所示：

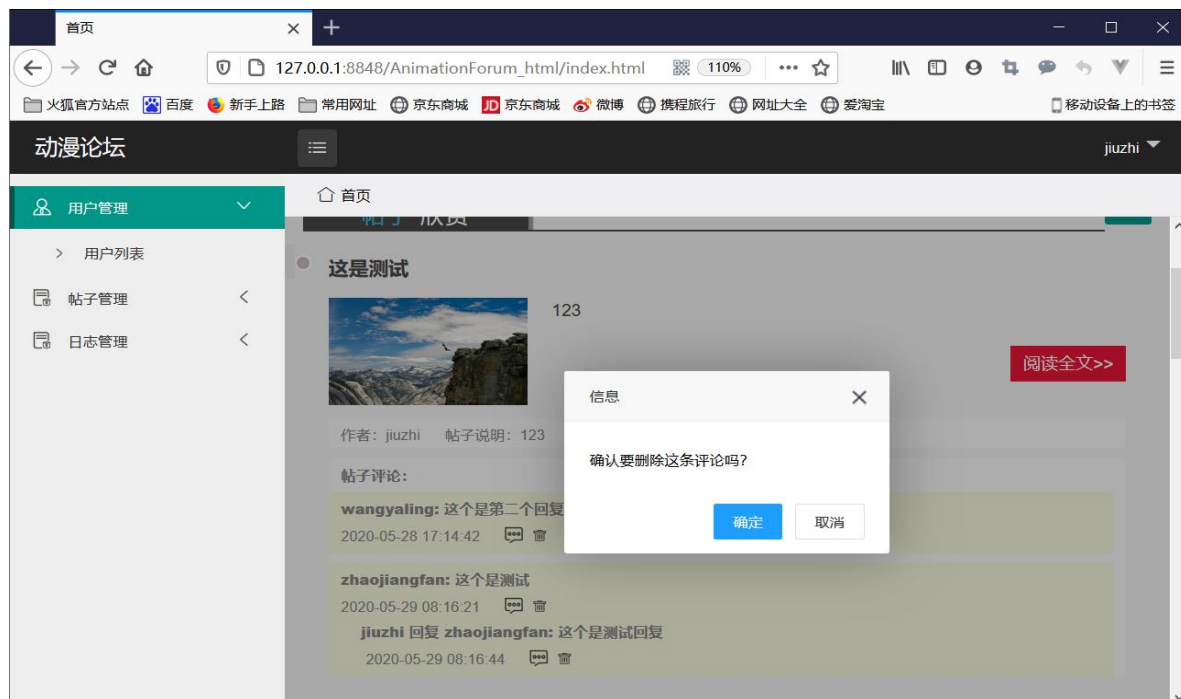


图 5.13 删除评论

用户在首页上点击删除回复按钮后，系统会弹出一个是否删除回复的提醒框，用户在提醒框上面进行确认删除操作或者取消删除操作，具体帖子删除回复的基本使用信息如图 5.14 所示：

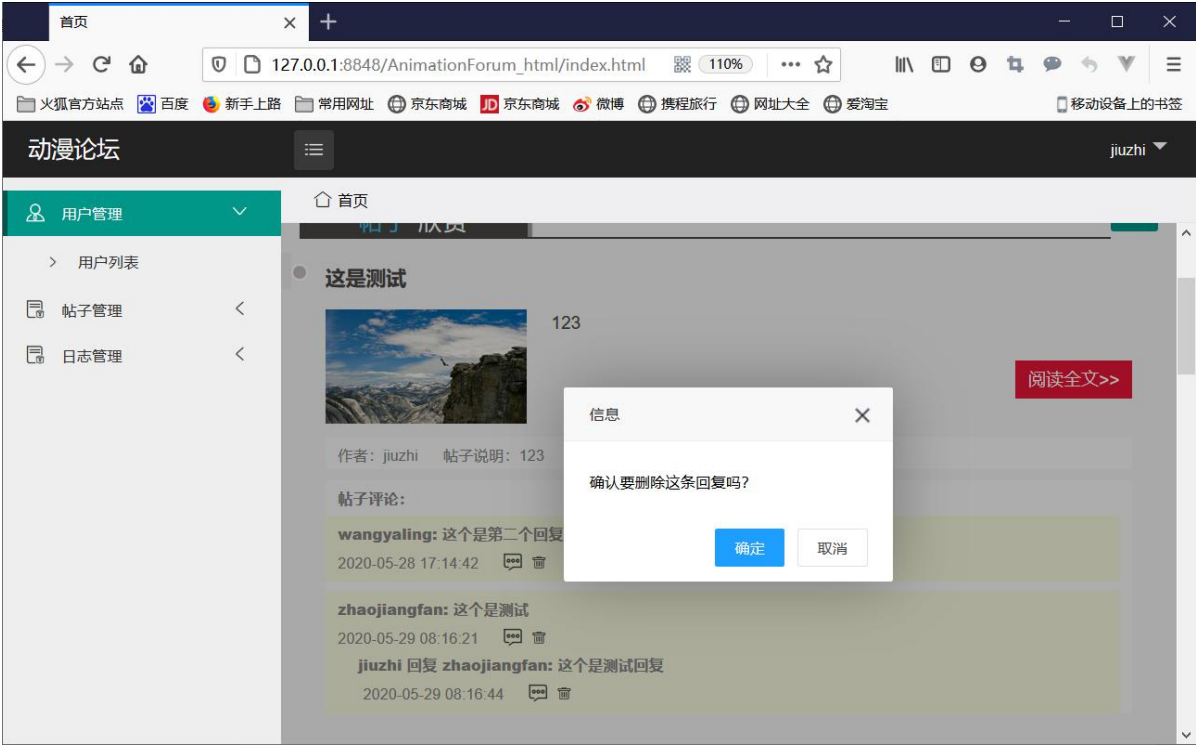


图 5.14 删除回复

至此，本文内容已经全部结束，感谢您的阅读。