**浙江大学宁波理工学院**

**2016-2017学年第1学期**

**《数据库系统技术及应用》大作业**



开课分院： 信息工程与科学学院

课程设计名称：社团活动登记系统

小组成员： 陈志鹏

完成日期： 2016-01-07

目 录

[1. 应用背景 3](#_Toc312411895)

[1.1 背景说明 3](#_Toc312411896)

[1.2 开发说明 3](#_Toc312411897)

[2. 需求分析 3](#_Toc312411898)

[2.1 系统目标 3](#_Toc312411899)

[2.2 需求描述 3](#_Toc312411900)

[2.3 业务过程 3](#_Toc312411901)

[2.4 功能分析 3](#_Toc312411902)

[3. 概念结构设计 4](#_Toc312411903)

[3.1 实体分析 4](#_Toc312411904)

[3.2 分ER图 4](#_Toc312411905)

[3.3 ER图集成 4](#_Toc312411906)

[4. 逻辑结构设计 4](#_Toc312411907)

[4.1 关系模式 4](#_Toc312411908)

[4.2 关系表设计 4](#_Toc312411909)

[4.3 视图设计 4](#_Toc312411910)

[4.4 索引设计 4](#_Toc312411911)

[5. 物理实现 5](#_Toc312411912)

[5.1 数据库的SQL脚本 5](#_Toc312411913)

[5.2 触发器 5](#_Toc312411914)

[5.3 存储过程 5](#_Toc312411915)

[5.4 用户自定义函数 5](#_Toc312411916)

[6. 应用开发 5](#_Toc312411917)

[6.1 系统部署 5](#_Toc312411918)

[6.2 系统展示 5](#_Toc312411919)

[7. 设计总结 5](#_Toc312411920)

[8. 附录 6](#_Toc312411921)

# 应用背景

## 背景说明

随着学校的蓬勃发展，学校的社团数量也随之变多。每个社团都会不定期举行社团活动，而每次社团活动都需要至学校有关部门进行审核，巨量的社团活动审核大大增加了学校有关部门的文书工作工作量，耗费了大量人力物力。

学校内网已经非常发达，因此我们想到利用网络构建一套基于Web的 社团活动登记系统 ，通过在线创建社团、提交社团活动申请的方式，有效降低有关部门对此的人力支出。同时，信息化的社团活动申请过程有助于社团活动历史的查询和数据统计。

## 开发说明

软件名称： 社团活动登记系统；

开发者： 陈志鹏；

# 需求分析

## 系统目标

系统目标：

构建一个基于 B/S 架构的在线社团活动登记系统，为学生和学校社团管理部门提供有效的社团活动管理平台。用于社团的创建，社团信息的修改，社团活动的申请和审核，学生参与社团活动的申请和审核等。

开发要求：

1. 系统名称： 社团活动登记系统
2. 开发语言 C#
3. 数据库管理工具： Mysql WorkBench， Aliyun DMS
4. 开发工具： IntelliJ Rider， Intellij DataGrip， Intellij dotTrace， Git-cli， Intellij TimeCity
5. 运行平台： .Net Core 1.0.1 及以上
6. 目标操作系统： Ubuntu 16.04 及以上

## 需求描述

学生：

申请创建社团；

对自己创建的社团修改信息；

加入社团；

为自己创建的社团申请进行社团活动；

参与自己加入的社团中已被审核通过的活动；

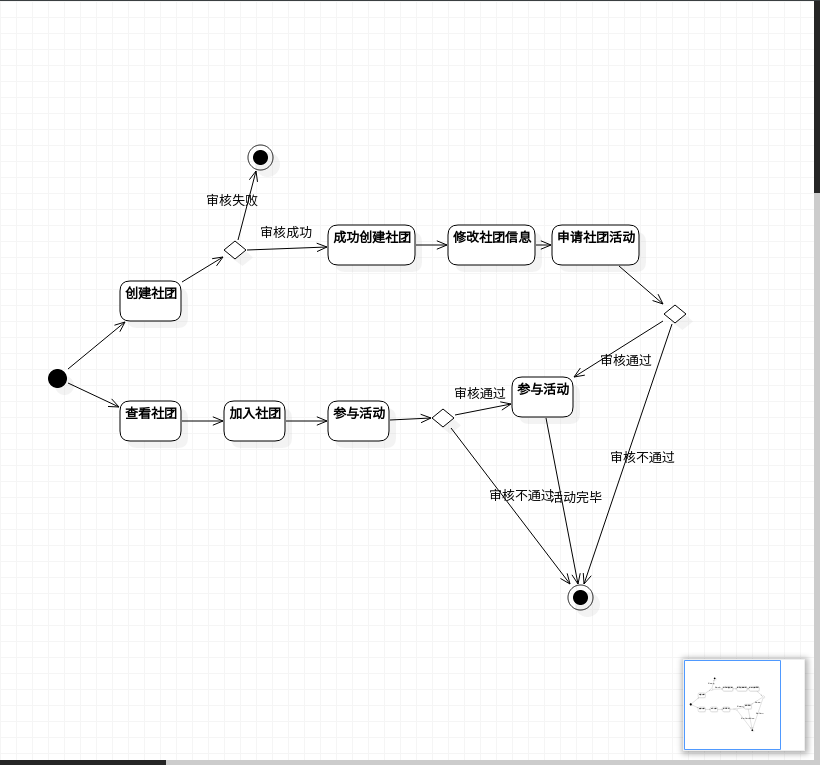
管理员：

对社团申请进行审核；

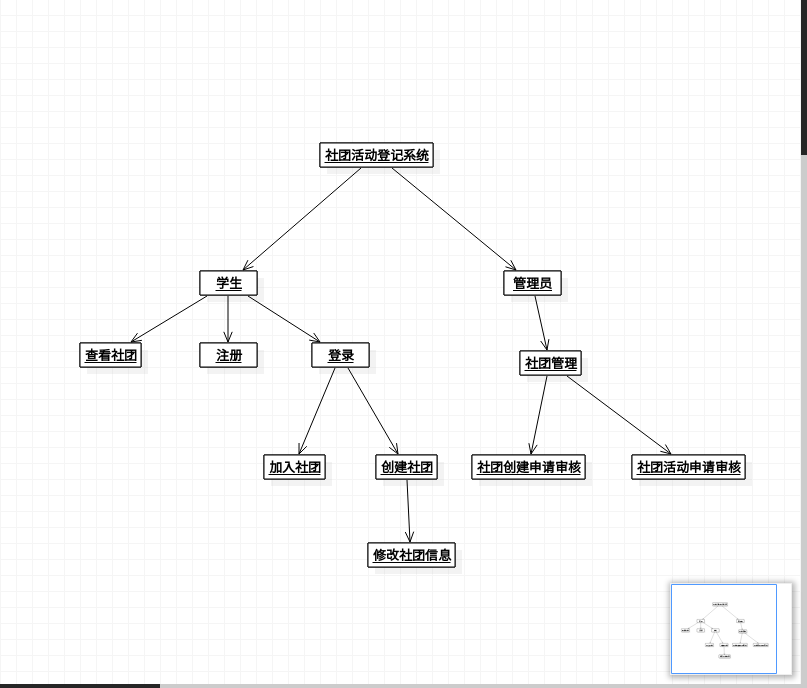
对社团活动申请进行审核；

查看所有社团活动历史和统计；

## 业务过程



## 功能分析



1. 普通学生

|  |  |  |
| --- | --- | --- |
| 功能名称 | 功能描述 | 备注 |
| 注册 | 注册一个新用户 | 已有用户名拒绝注册 |
| 登录 | 已注册用户使用注册信息登录 | 用户名、密码不匹配拒绝登录 |
| 查看社团 | 获得已有社团列表 |  |
| 创建社团 | 申请创建一个新社团 | 不可与已有社团重名 |
| 加入社团 | 加入一个已有社团 | 一个学生可以同时加入多个社团 |

1. 社长

|  |  |  |
| --- | --- | --- |
| 功能名称 | 功能描述 | 备注 |
| 修改社团信息 | 对自己创建的社团信息进行修改 |  |
| 发布新活动 | 申请进行社团活动 |  |

1. 管理员

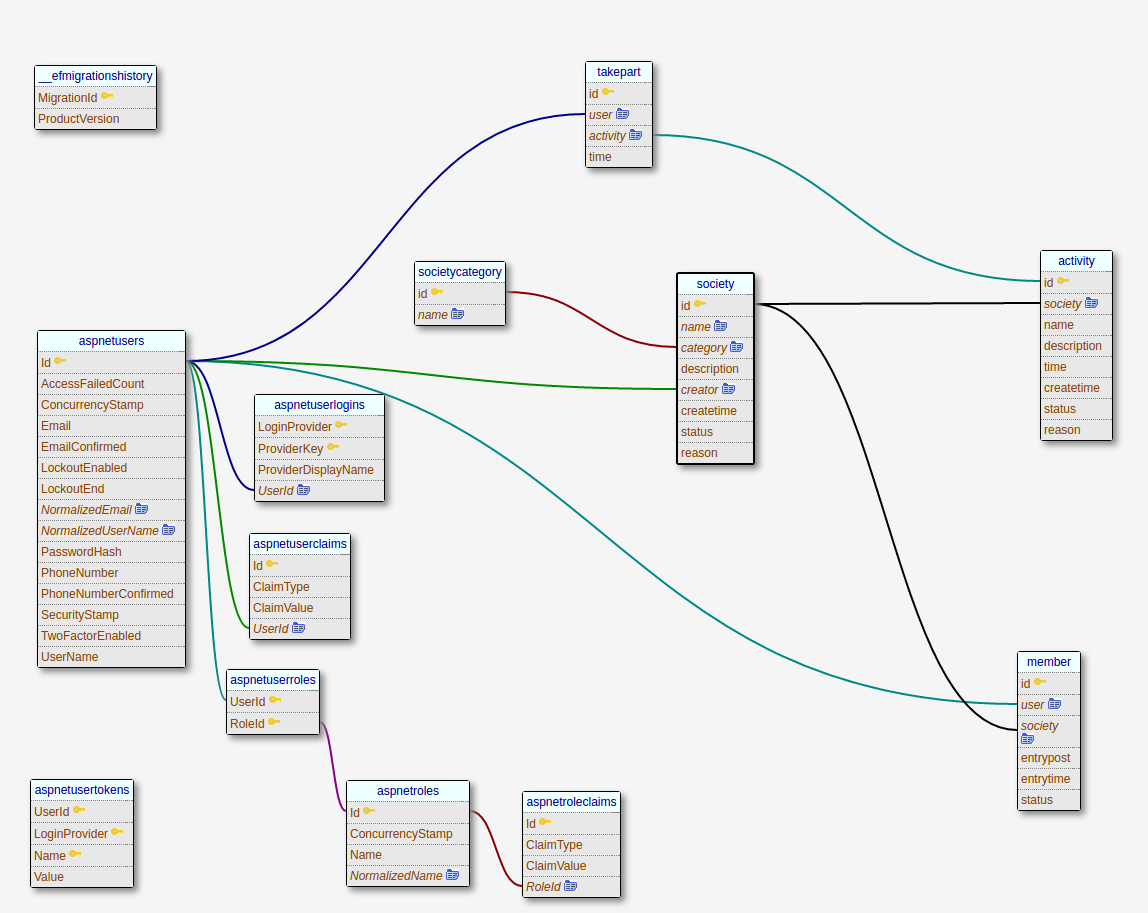
|  |  |  |
| --- | --- | --- |
| 功能名称 | 功能描述 | 备注 |
| 社团创建申请审核 | 对社团创建申请进行审核 | 审核后社团方可正常招人，申请活动 |
| 社团活动申请审核 | 对社团活动申请进行审核 | 审核后该活动方可进行 |

# 概念结构设计

## 实体分析

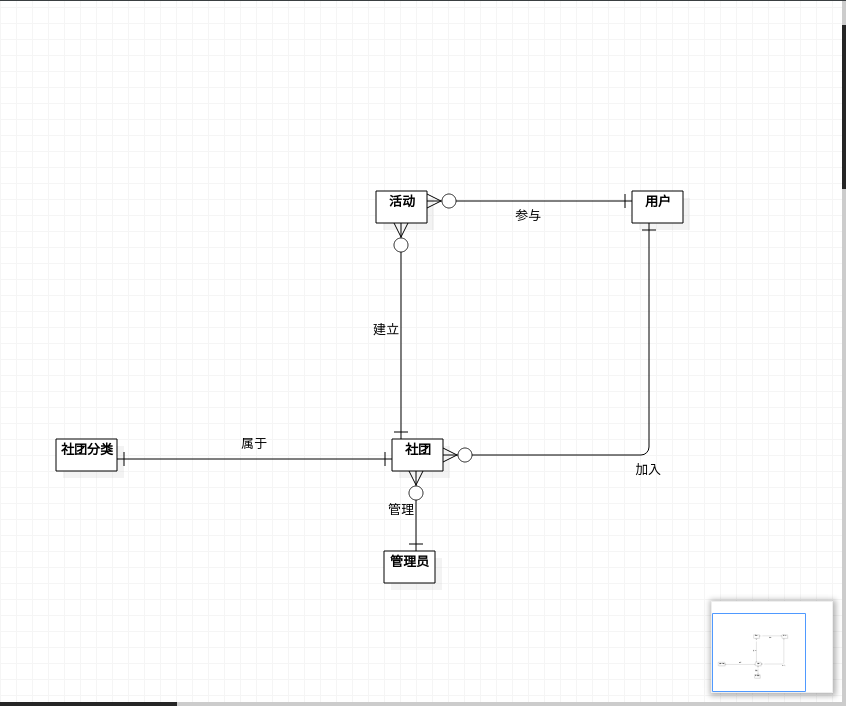
|  |  |  |
| --- | --- | --- |
| 序号 | 实体名称 | 举例描述及其属性 |
| 1 | 活动 | 公益献血（id，社团，活动名称，描述，时间，状态） |
| 2 | 管理员 | czp（id，用户名，密码） |
| 3 | 成员 | czp（id，用户名，社团） |
| 4 | 社团 | 插科打诨社（id，名称，分类，描述，创建者，状态） |
| 5 | 社团分类 | 文学类（id，名称） |
| 6 | 参与活动 | 参与活动（id，用户编号，活动编号） |
| 7 | 用户 | czp（id，用户名，密码） |

## 分ER图



（包含 .Net Core Identity 数据表）

## ER图集成



# 逻辑结构设计

## 关系模式

1. 活动（id，社团，名称，描述，时间，状态）； FK：社团
2. 管理员（id，用户名，密码）；
3. 成员（id，用户，社团）； FK：用户，社团
4. 社团（id，名称，分类，描述，创建者，状态）； FK：分类，创建者
5. 社团种类（id，名称）；
6. 参与（id，用户，活动）； FK：用户，活动
7. 用户（id，用户名，密码）；

## 关系表设计

**1、\_\_efmigrationshistory**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | MigrationId | varchar(45) | NOT NULL |  | -- |
| 2 | ProductVersion | varchar(45) | NOT NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | PRIMARY | PRIMARY | MigrationId |

**2、activity**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | int(11) **auto\_increment** | NOT NULL |  | -- |
| 2 | society | int(11) | NOT NULL |  | -- |
| 3 | name | varchar(45) | NOT NULL |  | -- |
| 4 | description | varchar(45) | NULL |  | -- |
| 5 | time | datetime | NOT NULL |  | -- |
| 6 | createtime | datetime | NULL | CURRENT\_TIMESTAMP | -- |
| 7 | status | varchar(45) | NULL | Pending | -- |
| 8 | reason | text | NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | index\_activity\_society | UNIQUE KEY | society |
| 2 | PRIMARY | PRIMARY | id |

**3、aspnetroleclaims**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | Id | int(11) | NOT NULL |  | -- |
| 2 | ClaimType | varchar(255) | NULL |  | -- |
| 3 | ClaimValue | varchar(255) | NULL |  | -- |
| 4 | RoleId | varchar(255) | NOT NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | IX\_AspNetRoleClaims\_RoleId | UNIQUE KEY | RoleId |
| 2 | PRIMARY | PRIMARY | Id |

**4、aspnetroles**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | Id | varchar(255) | NOT NULL |  | -- |
| 2 | ConcurrencyStamp | varchar(255) | NULL |  | -- |
| 3 | Name | varchar(255) | NULL |  | -- |
| 4 | NormalizedName | varchar(255) | NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | PRIMARY | PRIMARY | Id |
| 2 | RoleNameIndex | UNIQUE KEY | NormalizedName |

**5、aspnetuserclaims**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | Id | int(11) | NOT NULL |  | -- |
| 2 | ClaimType | varchar(255) | NULL |  | -- |
| 3 | ClaimValue | varchar(255) | NULL |  | -- |
| 4 | UserId | varchar(255) | NOT NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | IX\_AspNetUserClaims\_UserId | UNIQUE KEY | UserId |
| 2 | PRIMARY | PRIMARY | Id |

**6、aspnetuserlogins**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | LoginProvider | varchar(255) | NOT NULL |  | -- |
| 2 | ProviderKey | varchar(255) | NOT NULL |  | -- |
| 3 | ProviderDisplayName | varchar(255) | NULL |  | -- |
| 4 | UserId | varchar(255) | NOT NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | IX\_AspNetUserLogins\_UserId | UNIQUE KEY | UserId |
| 2 | PRIMARY | PRIMARY | LoginProvider , ProviderKey |

**7、aspnetuserroles**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | UserId | varchar(255) | NOT NULL |  | -- |
| 2 | RoleId | varchar(255) | NOT NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | IX\_AspNetUserRoles\_RoleId | UNIQUE KEY | RoleId |
| 2 | IX\_AspNetUserRoles\_UserId | UNIQUE KEY | UserId |
| 3 | PRIMARY | PRIMARY | UserId , RoleId |

**8、aspnetusers**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | Id | varchar(255) | NOT NULL |  | -- |
| 2 | AccessFailedCount | int(11) | NOT NULL |  | -- |
| 3 | ConcurrencyStamp | varchar(255) | NULL |  | -- |
| 4 | Email | varchar(255) | NULL |  | -- |
| 5 | EmailConfirmed | bit(1) | NOT NULL |  | -- |
| 6 | LockoutEnabled | bit(1) | NOT NULL |  | -- |
| 7 | LockoutEnd | timestamp **on update CURRENT\_TIMESTAMP** | NOT NULL | CURRENT\_TIMESTAMP | -- |
| 8 | NormalizedEmail | varchar(255) | NULL |  | -- |
| 9 | NormalizedUserName | varchar(255) | NULL |  | -- |
| 10 | PasswordHash | varchar(255) | NULL |  | -- |
| 11 | PhoneNumber | varchar(255) | NULL |  | -- |
| 12 | PhoneNumberConfirmed | bit(1) | NOT NULL |  | -- |
| 13 | SecurityStamp | varchar(255) | NULL |  | -- |
| 14 | TwoFactorEnabled | bit(1) | NOT NULL |  | -- |
| 15 | UserName | varchar(255) | NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | EmailIndex | UNIQUE KEY | NormalizedEmail |
| 2 | PRIMARY | PRIMARY | Id |
| 3 | UserNameIndex | UNIQUE KEY | NormalizedUserName |

**9、aspnetusertokens**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | UserId | varchar(255) | NOT NULL |  | -- |
| 2 | LoginProvider | varchar(255) | NOT NULL |  | -- |
| 3 | Name | varchar(255) | NOT NULL |  | -- |
| 4 | Value | varchar(255) | NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | PRIMARY | PRIMARY | UserId , LoginProvider , Name |

**10、member**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | int(11) **auto\_increment** | NOT NULL |  | -- |
| 2 | user | varchar(255) | NOT NULL |  | -- |
| 3 | society | int(11) | NOT NULL |  | -- |
| 4 | entrypost | text | NULL |  | -- |
| 5 | entrytime | datetime | NULL | CURRENT\_TIMESTAMP | -- |
| 6 | status | varchar(32) | NULL | Pending | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | index\_member\_society | UNIQUE KEY | society |
| 2 | index\_member\_user | UNIQUE KEY | user |
| 3 | PRIMARY | PRIMARY | id |

**11、society**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | int(11) **auto\_increment** | NOT NULL |  | -- |
| 2 | name | varchar(45) | NOT NULL |  | -- |
| 3 | category | int(11) | NOT NULL |  | -- |
| 4 | description | varchar(45) | NULL |  | -- |
| 5 | creator | varchar(255) | NOT NULL |  | -- |
| 6 | createtime | datetime | NULL | CURRENT\_TIMESTAMP | -- |
| 7 | status | varchar(45) | NULL | Pending | -- |
| 8 | reason | text | NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | index\_society\_category | UNIQUE KEY | category |
| 2 | index\_society\_creator | UNIQUE KEY | creator |
| 3 | index\_society\_name | UNIQUE KEY | name |
| 4 | PRIMARY | PRIMARY | id |

**12、societycategory**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | int(11) **auto\_increment** | NOT NULL |  | -- |
| 2 | name | varchar(45) | NOT NULL |  | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | index\_societycategory\_name | UNIQUE KEY | name |
| 2 | PRIMARY | PRIMARY | id |

**13、takepart**( InnoDB , utf8\_general\_ci ) **[返回顶部](https://dms-rds.aliyun.com/mysql/dictionary/showdict.do?nowDBName=dotnet_home_work&token=3bd5dc83-73d4-4617-bf47-f18dd1bcb7ee&panelKey=dictionary" \l "XTOP)**

| **序号** | **列名** | **数据类型** | **是否可空** | **默认值** | **注释** |
| --- | --- | --- | --- | --- | --- |
| 1 | id | int(11) **auto\_increment** | NOT NULL |  | -- |
| 2 | user | varchar(255) | NOT NULL |  | -- |
| 3 | activity | int(45) | NOT NULL |  | -- |
| 4 | time | datetime | NULL | CURRENT\_TIMESTAMP | -- |

| **序号** | **索引名** | **类型** | **包含字段** |
| --- | --- | --- | --- |
| 1 | index\_take\_part\_activity | UNIQUE KEY | activity |
| 2 | index\_take\_part\_user | UNIQUE KEY | user |
| 3 | PRIMARY | PRIMARY | id |

## 视图设计

1. 社团信息(id,社团名，社团分类名，社团描述，创建者id，创建者名称，状态)；
2. 活动信息(id，社团名，活动名，活动描述，时间，状态)；
3. 参与信息（id，用户名称，活动名称）；
4. 用户信息（id，用户名）；

详见目录下 sql 文件

## 索引设计

见关系表设计一节

# 物理实现

## 数据库的SQL脚本

见文档目录下 dotnet\_homework.sql 文件

或项目目录下 Migrations 文件

## 触发器

（设计应用相关的触发器，每个组员至少实现一个触发器）

## 存储过程

（实现若干个应用中的功能，每个组员至少实现一个存储过程）

## 用户自定义函数

CREATE FUNCTION `f\_CountOfSocietiesIJoined`(

in\_userId VARCHAR(255)

) RETURNS int(11)

DETERMINISTIC

COMMENT '获取我加入的社团数量'

BEGIN

DECLARE cnt INT ;

SELECT COUNT(\*) INTO cnt

FROM `member`

WHERE `user`= in\_userId

AND `status`= "Accept";

RETURN cnt;

END

# 应用开发

## 系统部署

1. 安装 .Net Core

详见 <https://www.microsoft.com/net/core#linuxubuntu>

命令行

# sudo sh -c 'echo "deb [arch=amd64] https://apt-mo.trafficmanager.net/repos/dotnet-release/ xenial main" > /etc/apt/sources.list.d/dotnetdev.list'

# sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 417A0893

# apt-get update

# sudo apt-get install dotnet-dev-1.0.0-preview2-003131

tip1：若安装不同版本 .Net Core，可能需要自行修改项目框架版本，见项目目录下 project.json

**其他操作系统请自行参照微软官方文档进行安装，此处不赘述**

2.获取项目代码

命令行

# git clone <https://github.com/czp3009/DotnetHomework>

3.补全依赖

详见 <https://docs.microsoft.com/en-us/dotnet/articles/core/tools/index>

命令行

# cd ./DotnetHomework/DotnetHomeWork

# dotnet restore

4.编译和运行

详见 <https://docs.microsoft.com/en-us/aspnet/core/getting-started>

命令行

# dotnet run

5.项目设定

方法一，使用环境变量

命令行

# export ASPNETCORE\_ENVIRONMENT=Development

# export ASPNETCORE\_URLS=http://0.0.0.0:5000

方法二，使用配置文件

见项目目录下 ./DotnetHomeWork/appsettings.json 文件

**默认将使用一台RDS服务器（存在 ip 白名单），若需使用自建数据库请继续查看接下来的步骤。**

6.数据库安装和数据表生成

执行以下步骤前，请首先修改数据库 ConnectionString ，见项目配置文件。

第一步，安装数据库

命令行

# apt install mysql-server

第二步，运行数据库

# service mysql start

第三步，生成数据表

方法一，使用 sql 文件

命令行

# mysql -u root -p

# create database dotnet\_home\_work;

# use dotnet\_home\_work;

# source {文档目录}/dotnet\_homework.sql

方法二，使用 Migrations 文件

命令行

# cd ./DotnetHomework

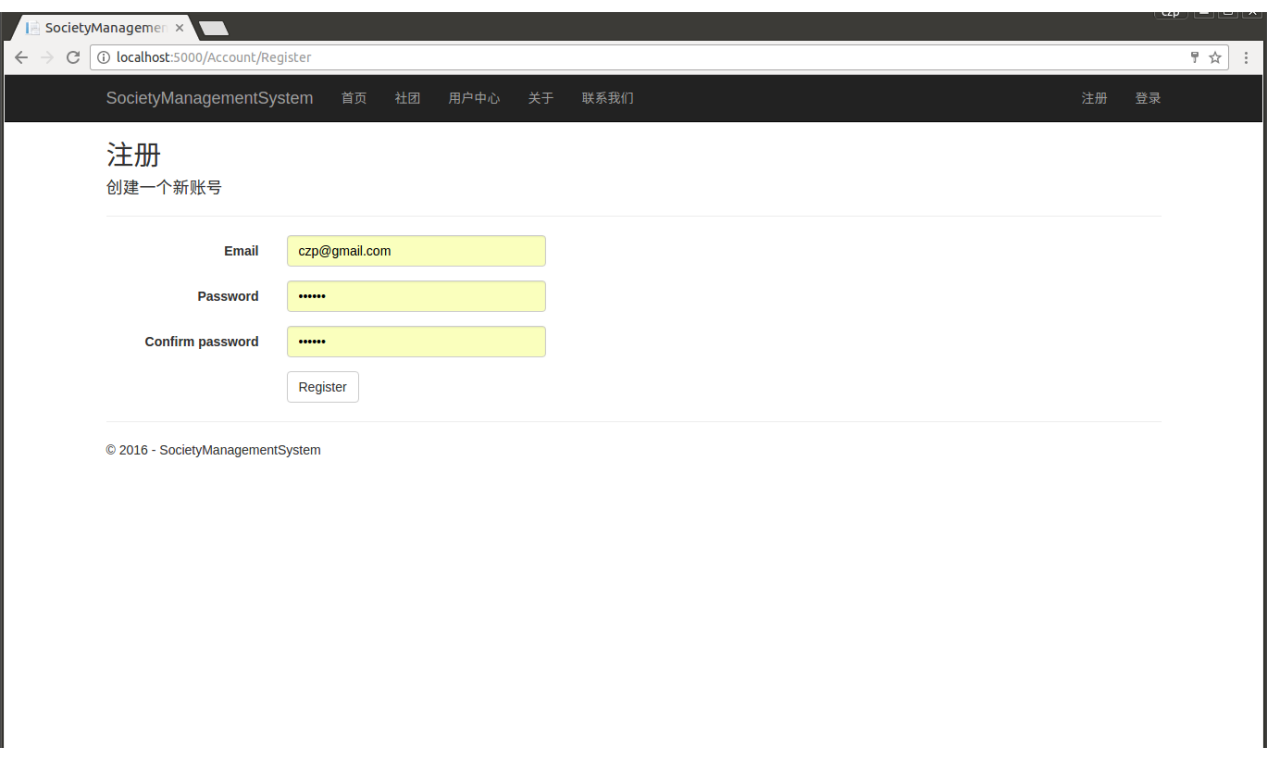
# dotnet ef database update -c ApplicationDbContext

# dotnet ef database update -c SocietyManagementSystemDbContext

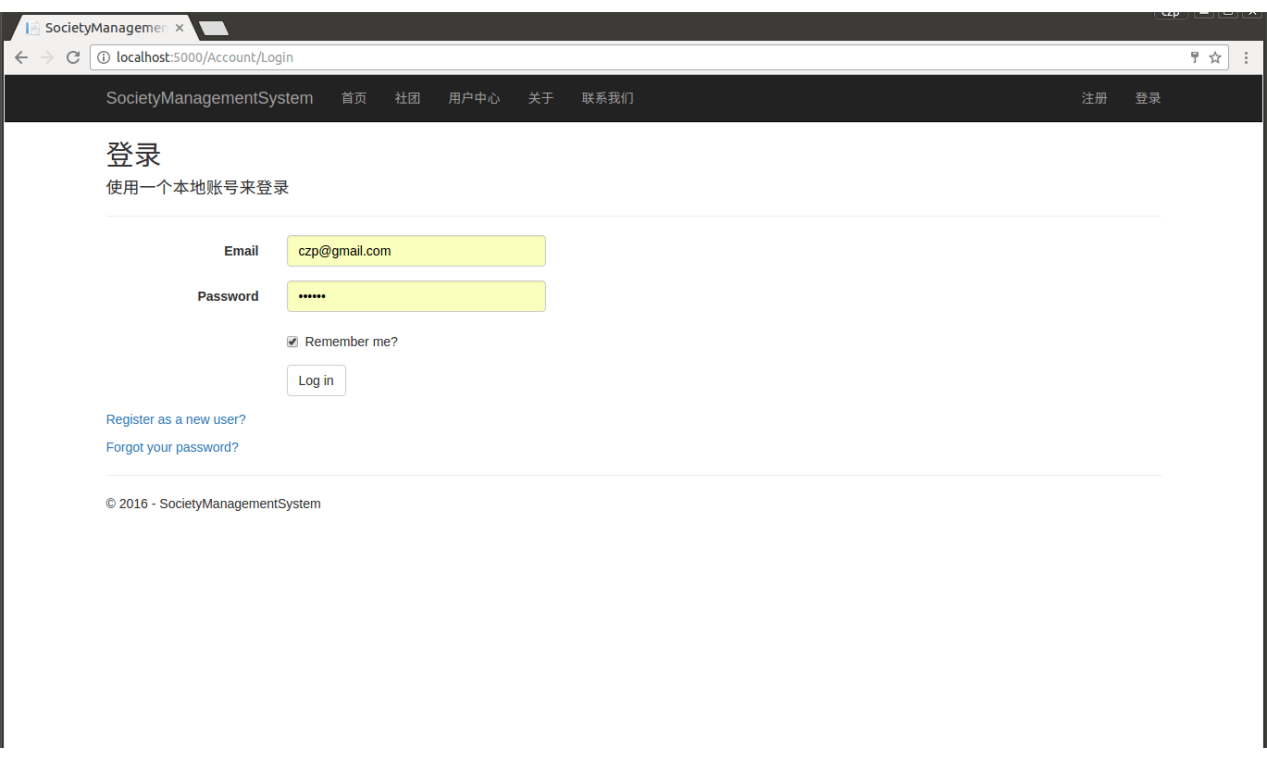
## 系统展示

Demo 演示见 <http://qd.hiczp.com/>

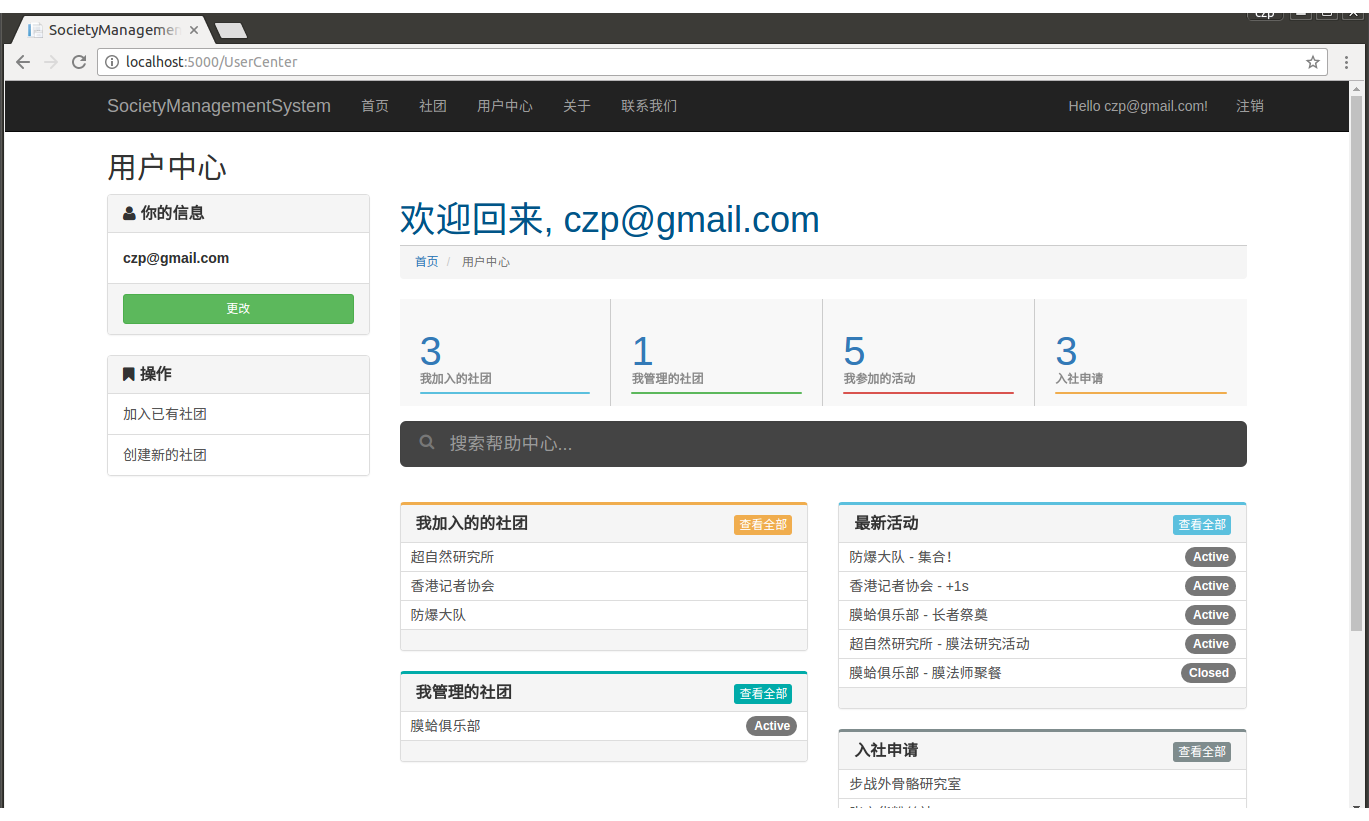
注册



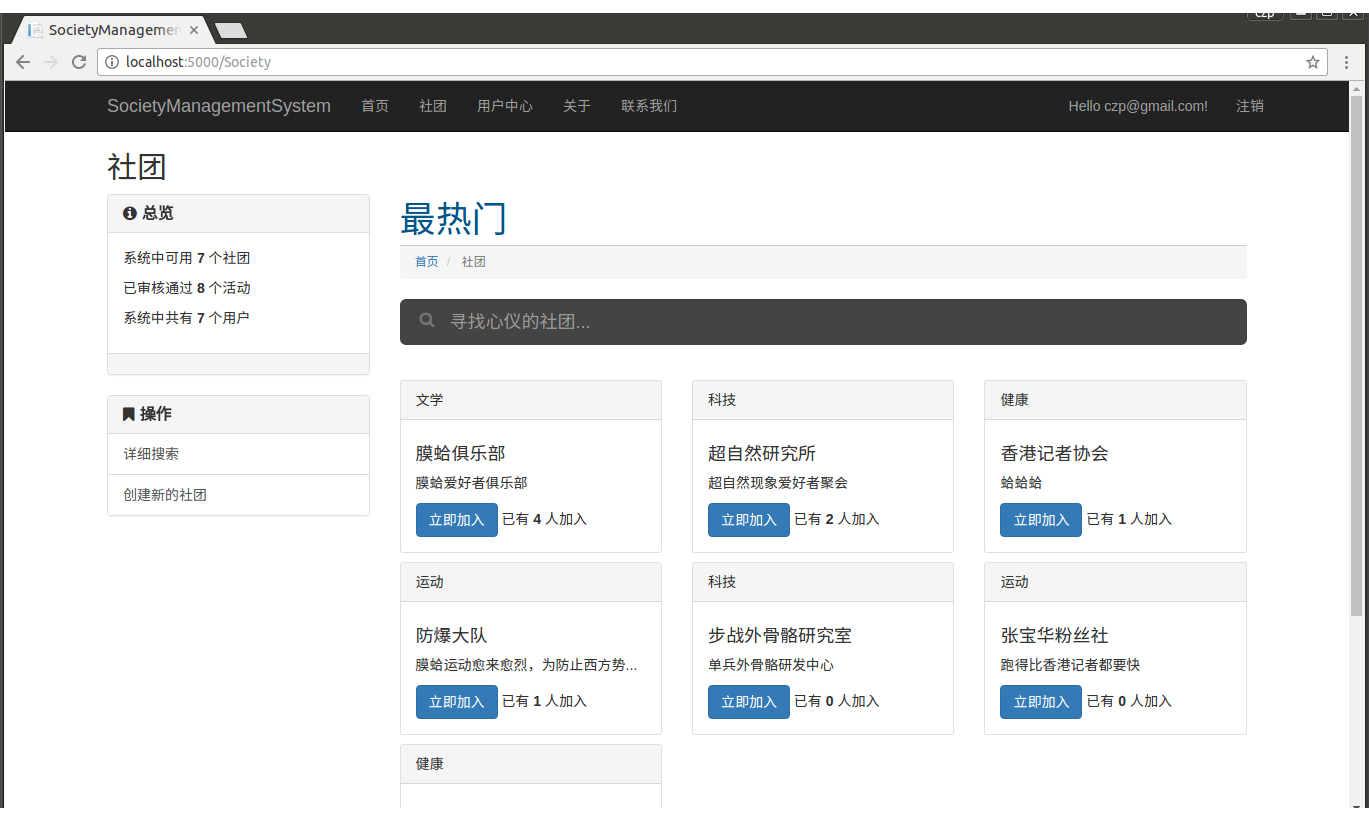
登录



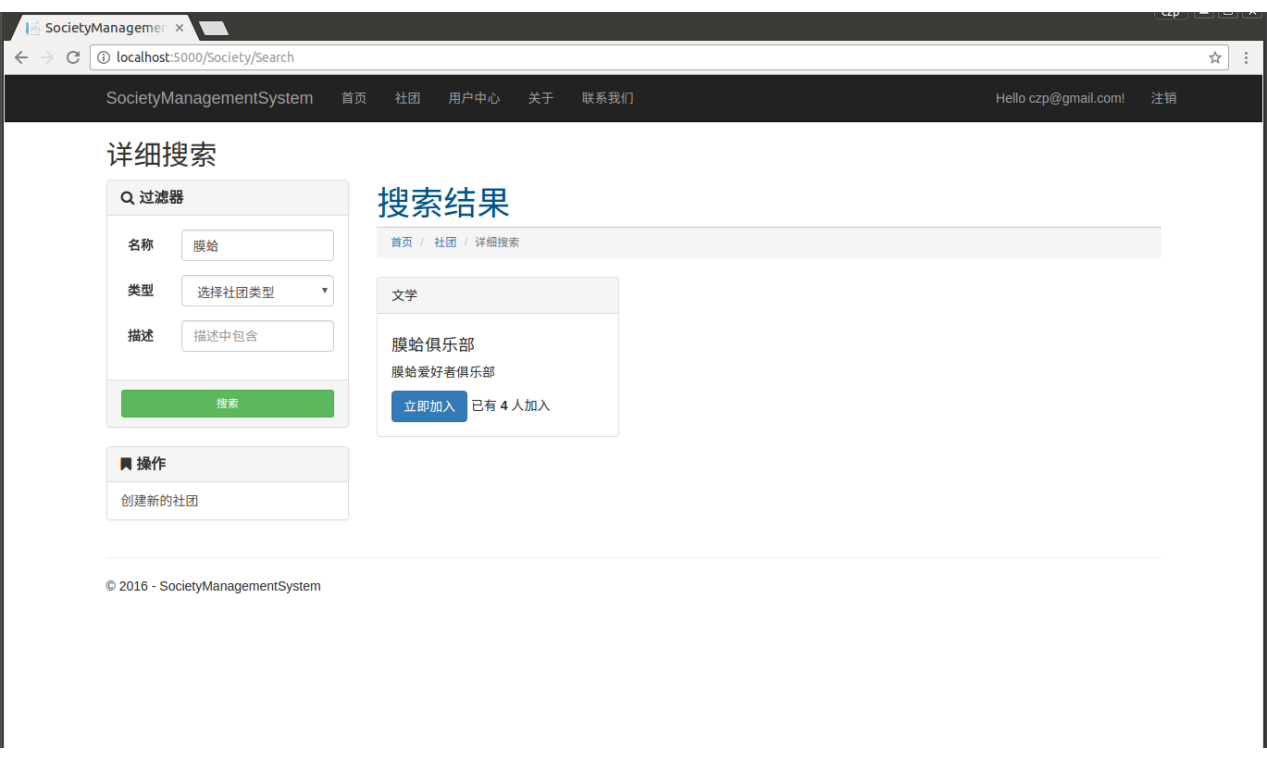
用户中心



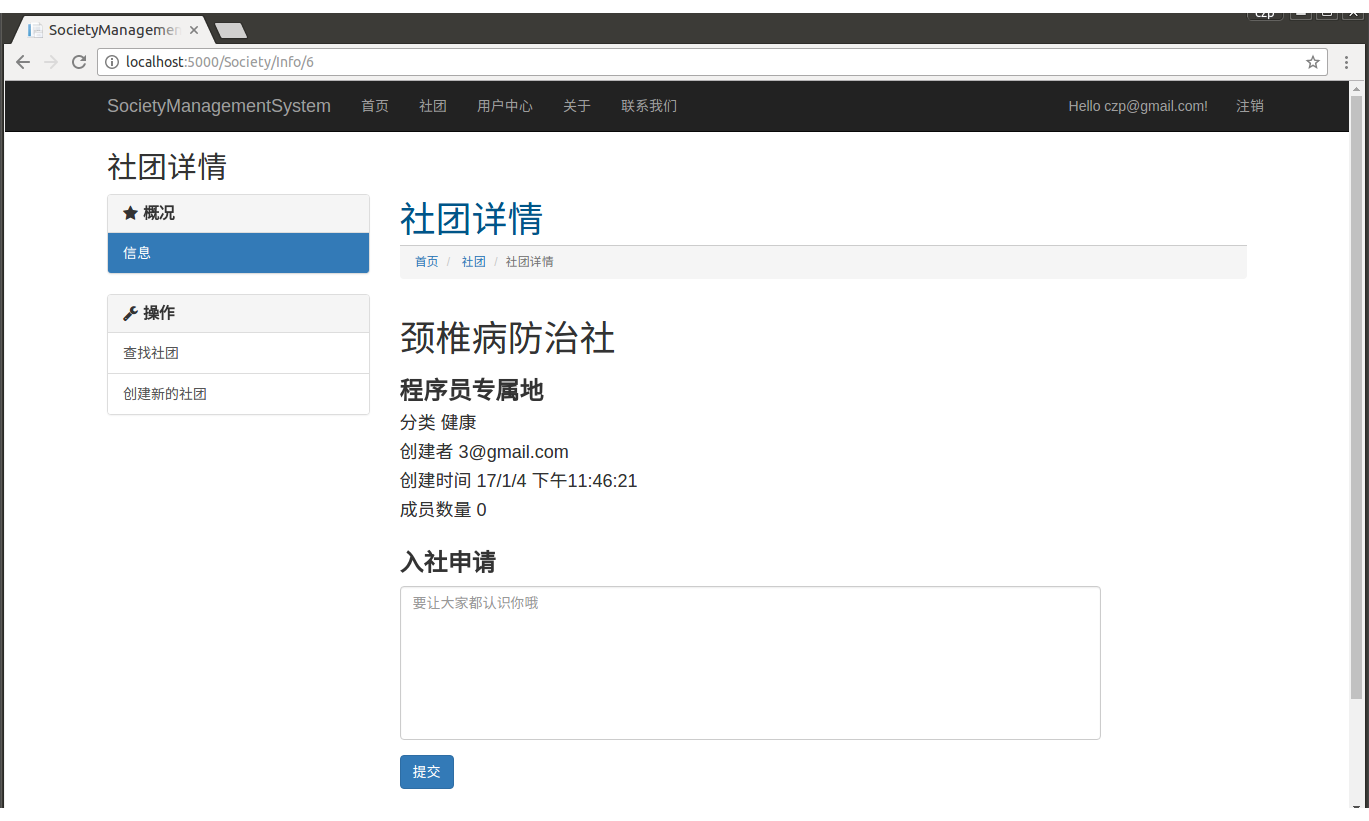
发现社团



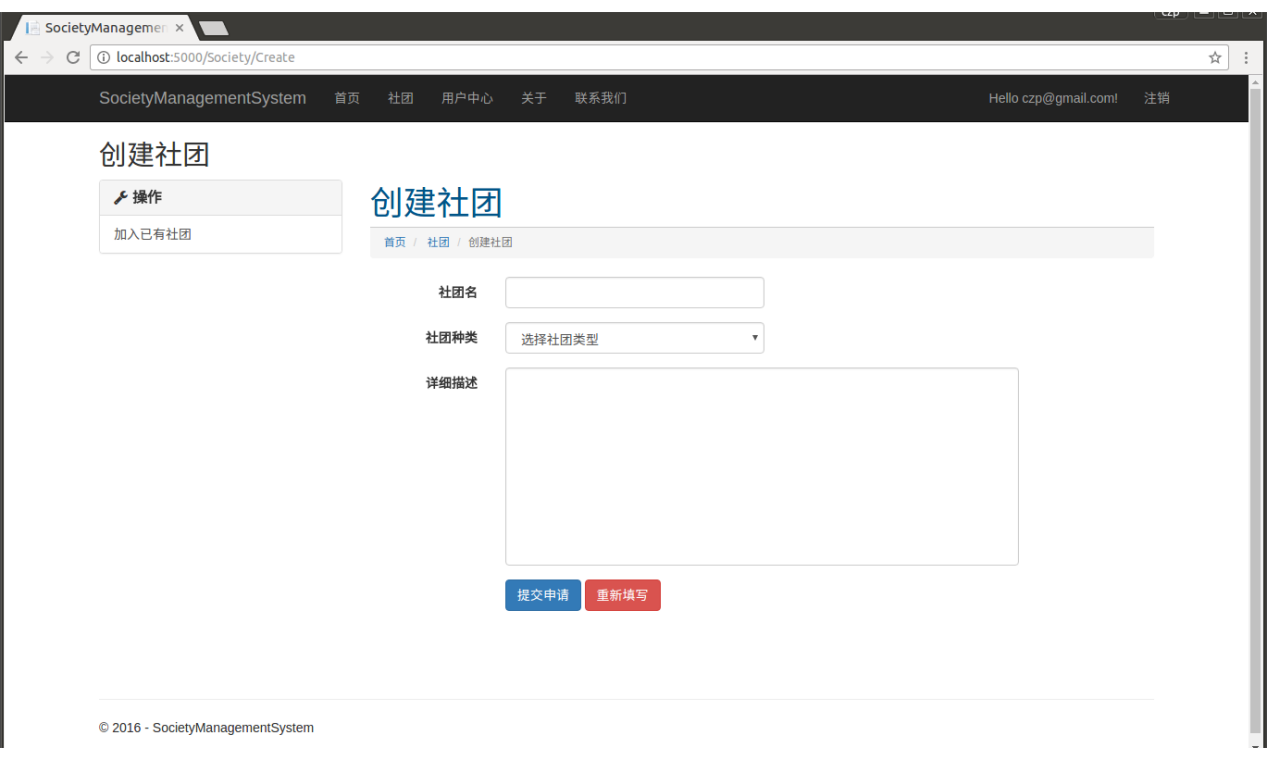
查找社团



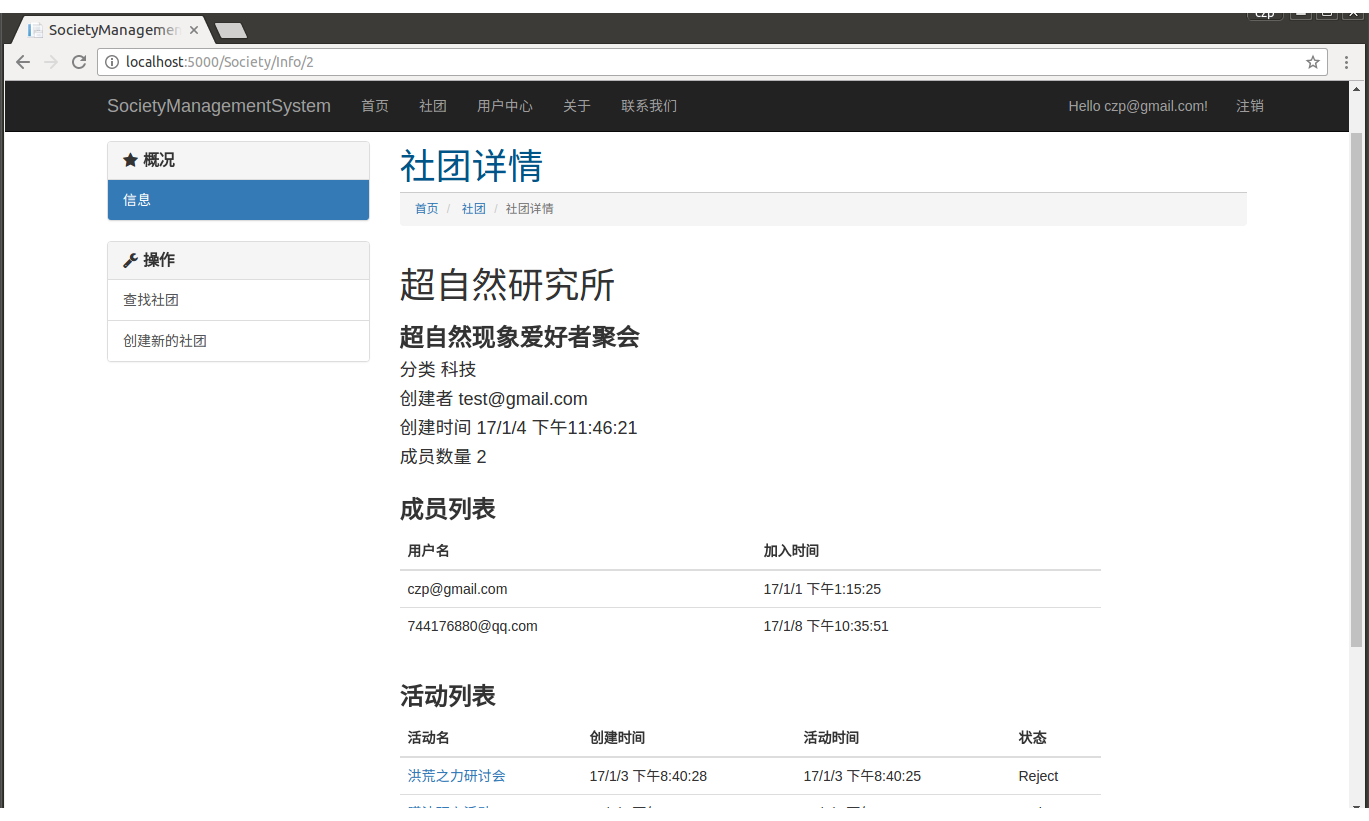
加入社团



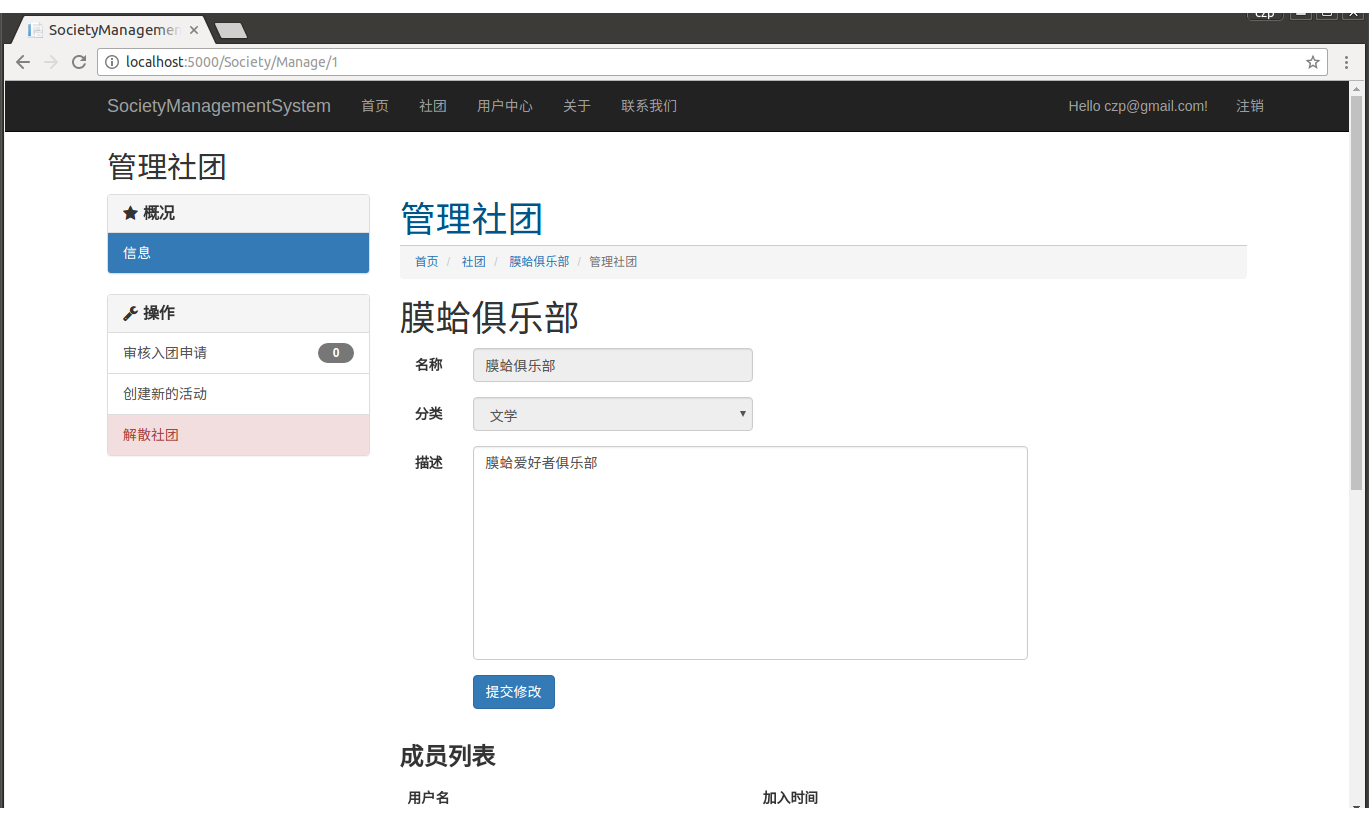
创建社团



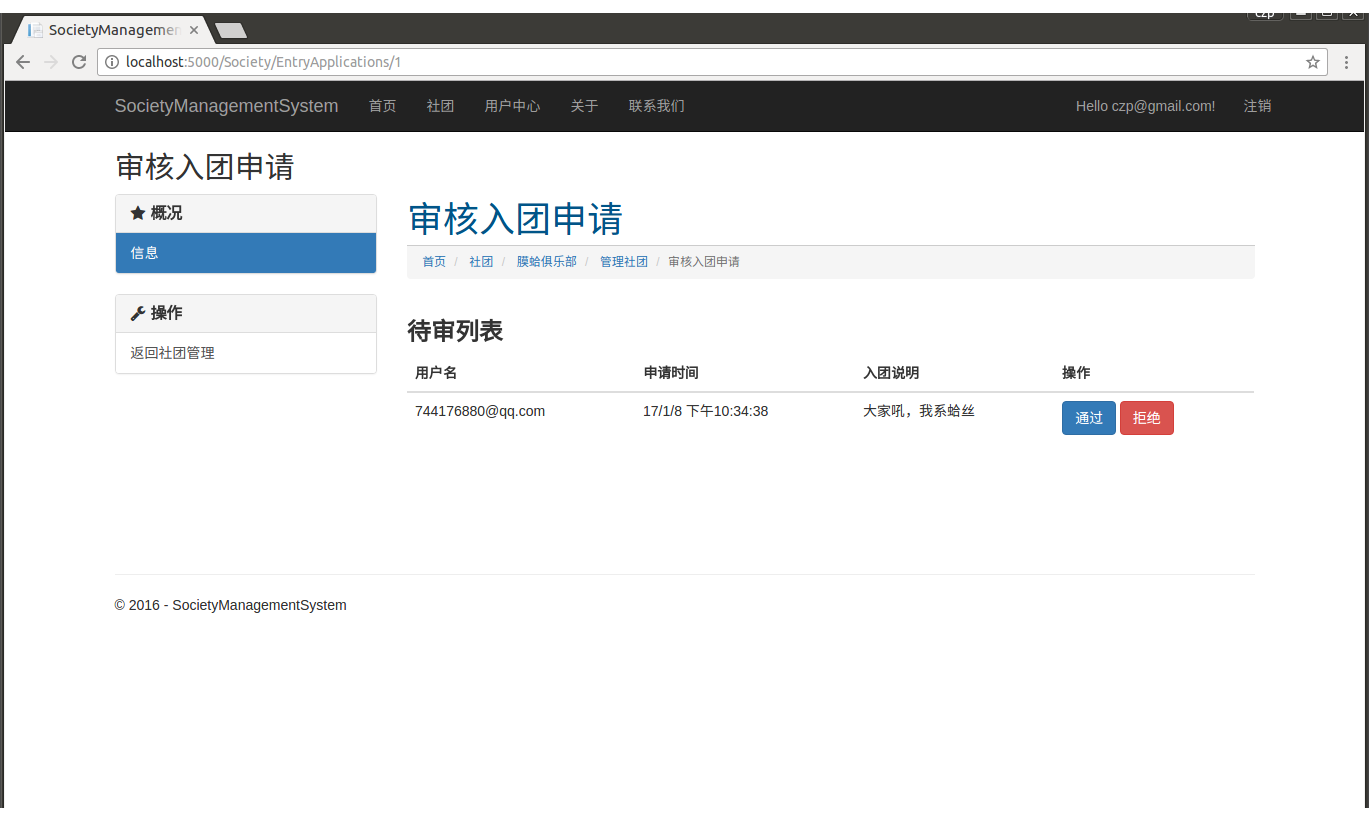
查看社团详情



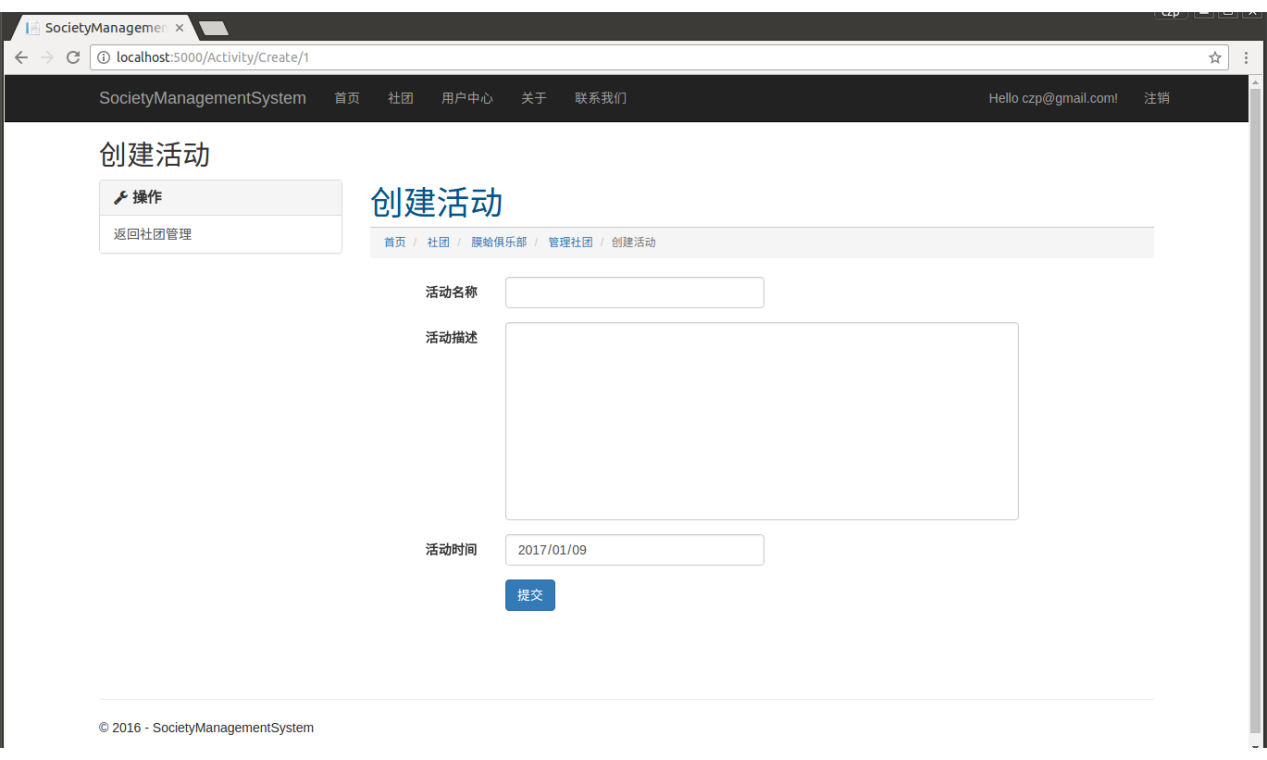
管理自己的社团



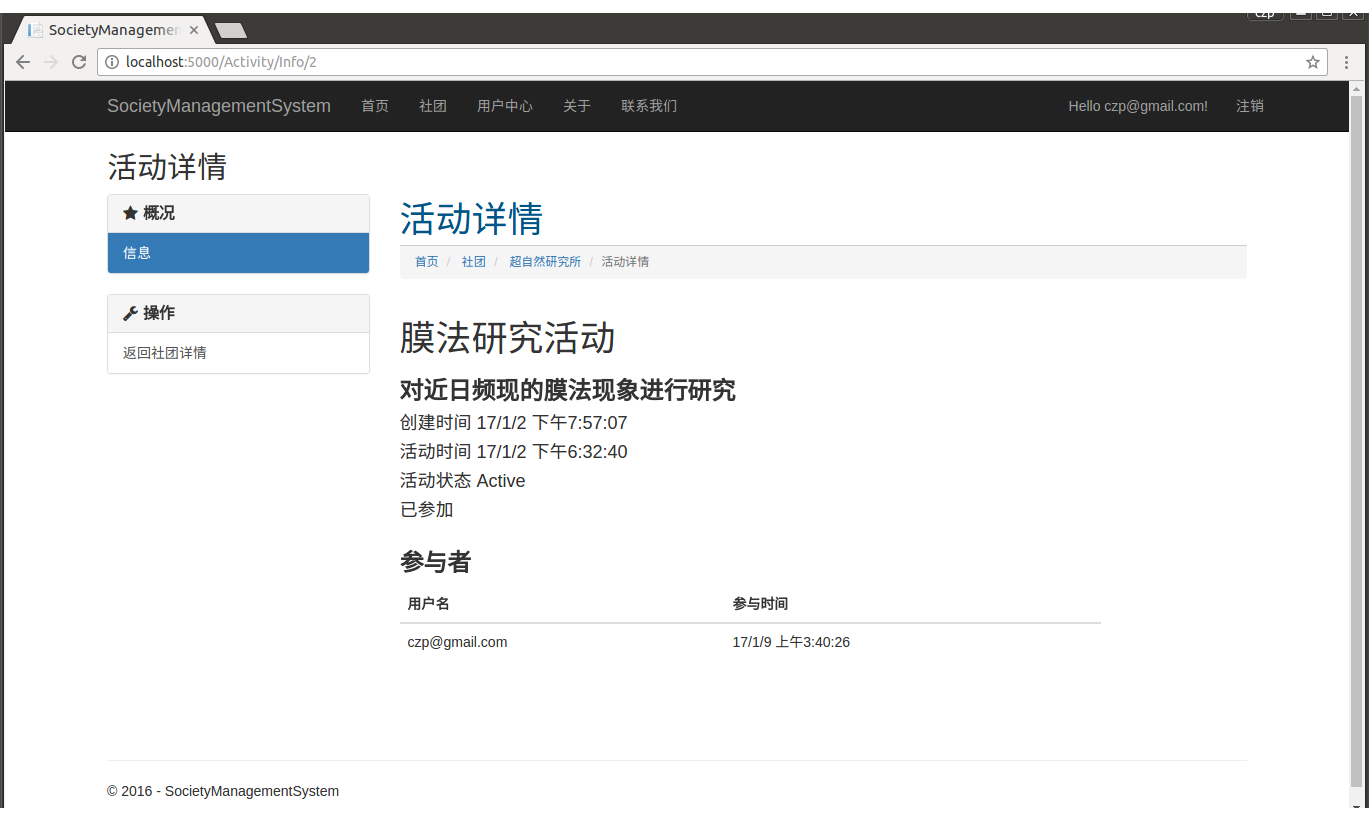
审核成员



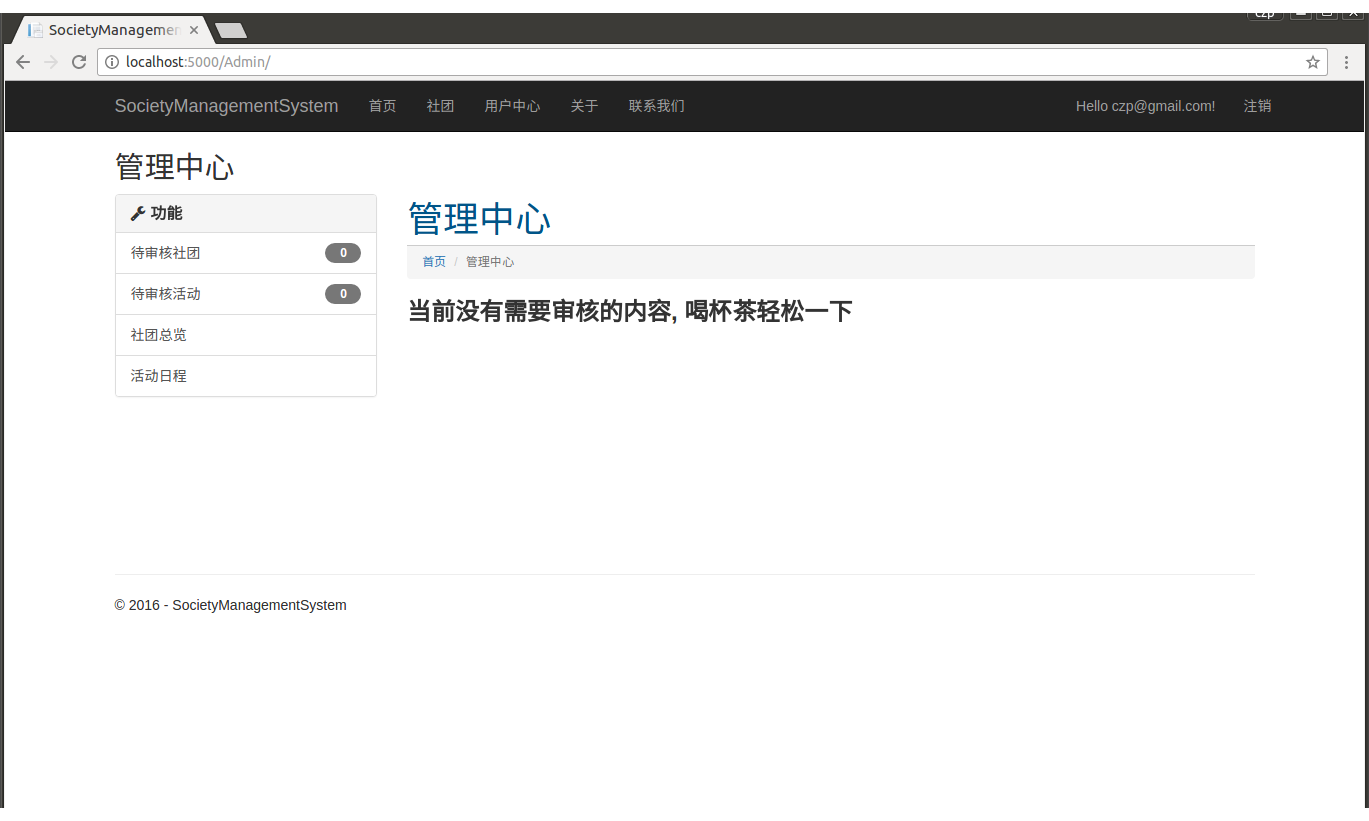
创建社团活动



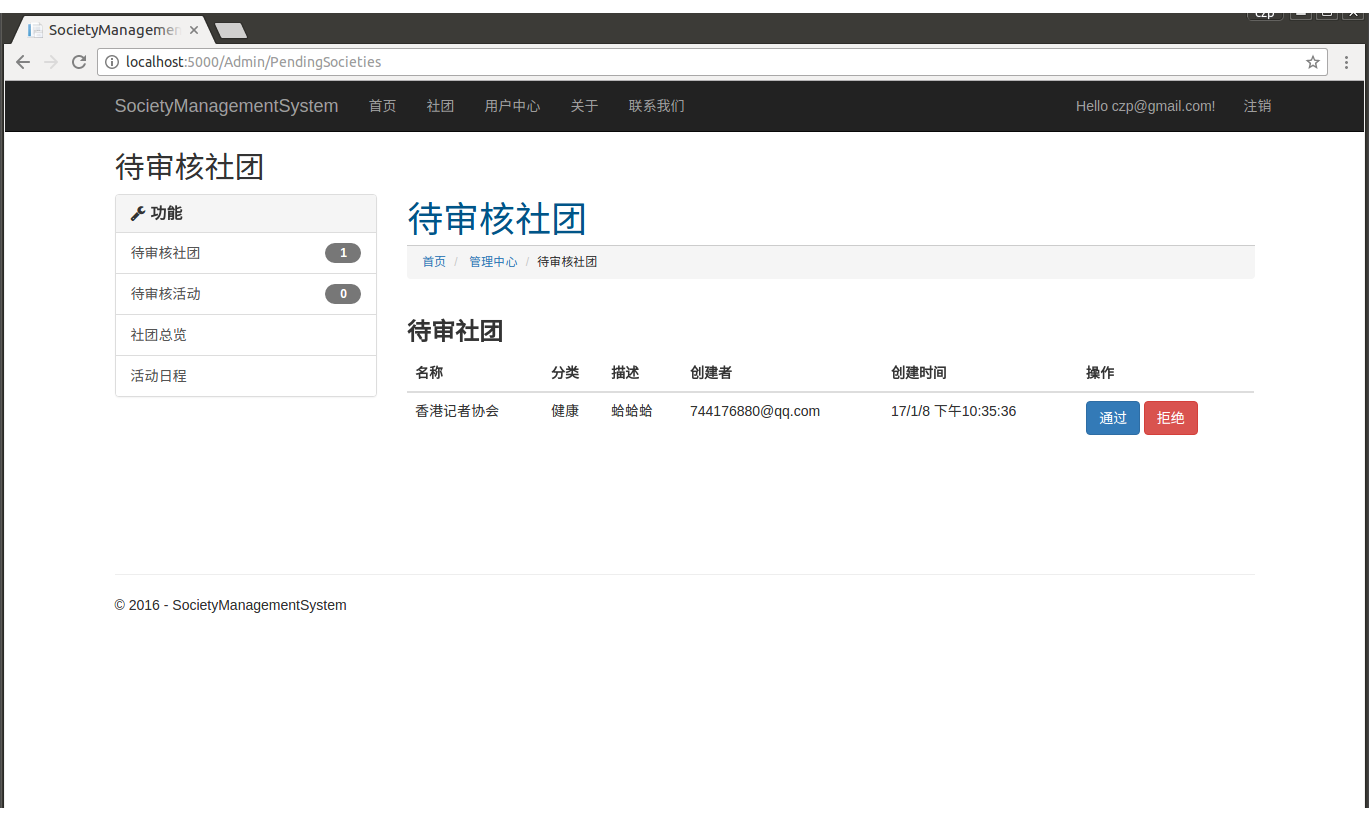
加入活动



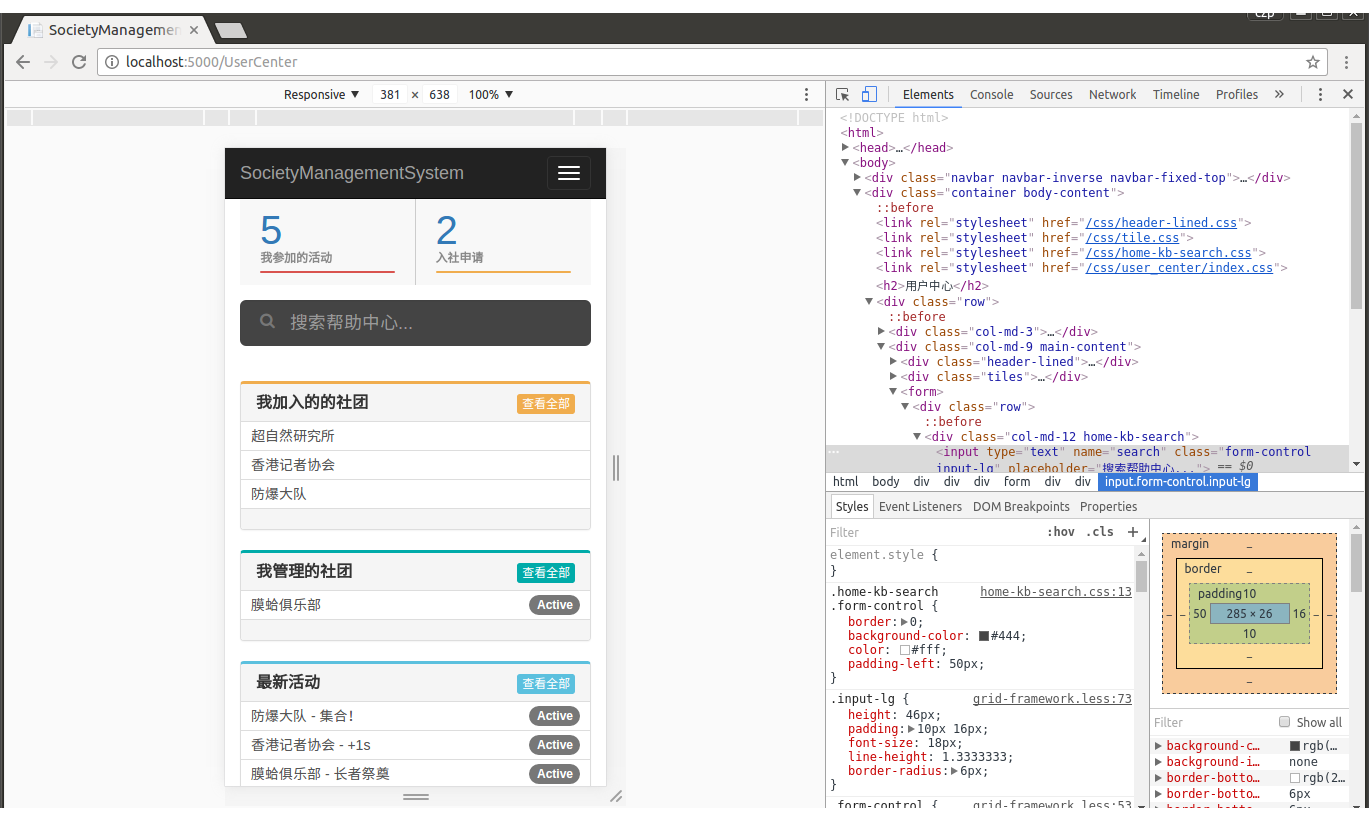
管理员功能



管理员审核社团与活动



智能终端响应式布局



# 设计总结

通过本次课设，在设计和实现过程中掌握了 .Net Core, ASP.NET Core, EntityFramework Core, 等多种应用技术。同时对 MySql 等数据库系统的使用和编程有了更深入理解，收益颇丰。

# 附录

详见 <https://github.com/czp3009/DotnetHomework>