

悦青春社团活动管理系统

[软件设计说明书]

目录

1. 引言	1
1.1. 编写目的	1
1.2. 术语定义	1
1.3. 相关文档	1
2. 整体设计	1
2.1. 硬件运行环境	1
2.2. 软件运行环境	2
2.3. 系统管理员	2
2.4. 社团成员	3
2.5. 社团管理员	3
3. 数据库设计	4
3.1. 数据库中表名列表	4
3.2. 数据库E-R图	4
3.3. 数据库各表具体字段	5
4. 典型功能子系统设计	8
4.1. MD5加密设计	8
4.2. UserID Base64加密	9

1. 引言

1.1. 编写目的

在完成了社团活动管理系统需求分析的基础上，我们依据《需求规格说明书》对我们的项目进行了整体上的设计，同样也是为了将我们项目的设计变得文档化才编写了这一软件设计说明书。

1.2. 术语定义

本系统：“社团活动管理系统”项目本身，包括文档和源代码。

1.3. 相关文档

- [1] 《需求规格说明书》
- [2] 《软件开发计划书》
- [3] 《部署文档》
- [4] 《测试报告》
- [5] 《用户使用说明书》

2. 整体设计

2.1. 硬件运行环境

处理器：Intel® Core™ i5-13490F CPU @2.50GHz

内存：32GB

硬盘容量：1TB

输入输出设备：鼠标、键盘、显示屏

网络设备：宽带

2.2. 软件运行环境

操作系统: Windows 11
连接的后端数据库: MySQL 8.0.40
项目运行环境: JDK 11
浏览器: Google Chrome
后端开发: SpringBoot
前端开发: vue/cli-service 4.5.0

2.3. 系统管理员

①用户管理

a. 注册用户账户

当一个新用户注册时, 用户填写基本信息并上传。用户基本信息包括账号、姓名、密码、手机、地址等信息。

b. 用户信息管理

管理员可以查看系统所有用户的基本信息, 并修改和删除。

②社团管理

a. 创建社团

社团由管理员创建, 创建社团需填写社团名称, 社团类型和社团社长。

b. 社团信息管理

管理员可以查看使用该系统的所有社团基本信息, 修改和删除该社团。由于需要处理的信息可能更多, 为了使页面更简洁, 减少数据库查询的压力, 并使管理员能够更快地找到需要处理的社团信息, 添加社团名称字段的查询功能

2.4. 社团成员

①用户管理

- a. 个人信息管理 用户可以查看自己的个人信息，也可以修改、删除操作自己不满意的信息。
- b. 申请加入社团 用户可以浏览所有社团，并向喜欢的社团提交申请。
- c. 加入活动 用户可以浏览加入社团的全部活动，并选择加入到感兴趣的活动中。
- d. 缴费信息管理 用户可以收到缴费通知，并进行缴费。

2.5. 社团管理员

①社团管理

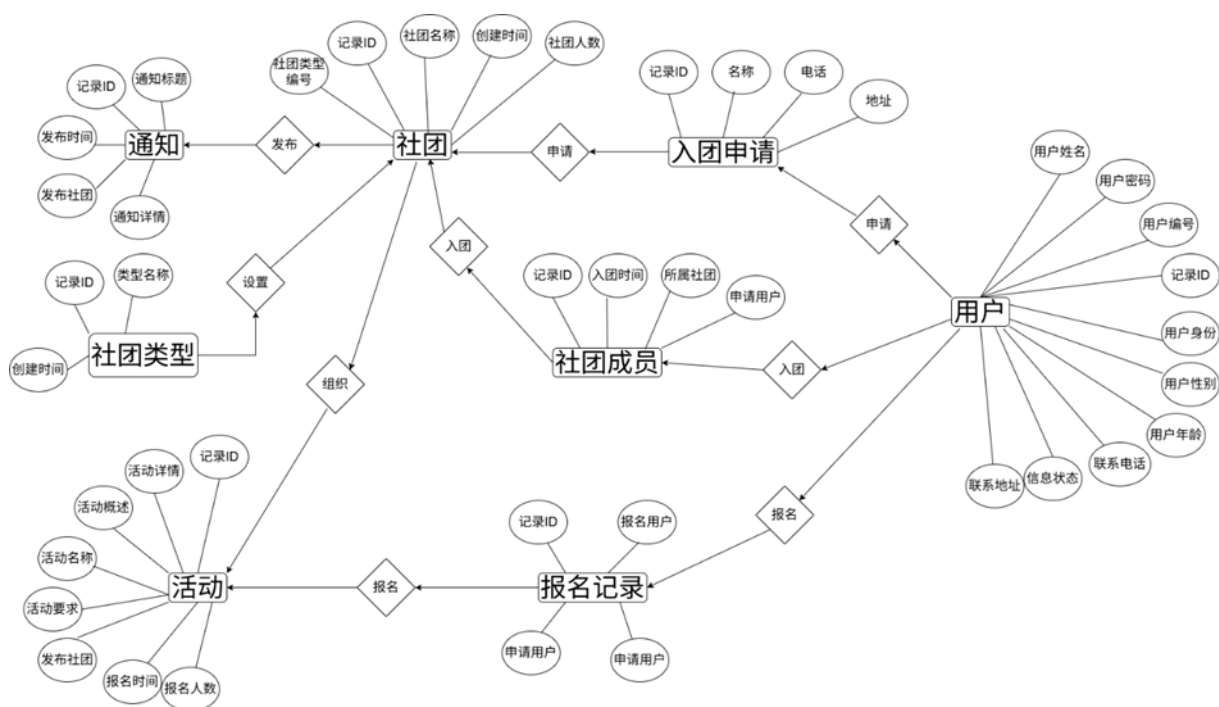
- a. 活动信息管理 社团管理员可以向所有社员发布活动信息，可以对参与活动的社员进行审核。
- b. 社团成员信息管理 社团管理员可以对社员进行删除，对入团申请者进行审核。
- c. 缴费信息管理 社团管理员可以根据社员编号发布缴费信息。

3. 数据库设计

3.1. 数据库中表名列表

名	自动...	修改日期	数据长...	引擎	行	注释
active_logs	0		16 KB	InnoDB	19	报名记录
activities	0		16 KB	InnoDB	16	活动信息
apply_logs	0		16 KB	InnoDB	18	申请记录
members	0		16 KB	InnoDB	29	成员信息
notices	0		16 KB	InnoDB	24	通知记录
pay_logs	0		16 KB	InnoDB	13	缴费记录
team_types	0		16 KB	InnoDB	26	社团类型
teams	0		16 KB	InnoDB	13	社团信息
users	0		16 KB	InnoDB	21	系统用户

3.2. 数据库E-R图



3.3. 数据库各表具体字段

表 4-3-1 用户表 (users)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
user_name	varchar	32		NOT NULL	用户账号
pass_word	varchar	32		NOT NULL	用户密码
name	varchar	20			用户姓名
gender	char	2			用户性别
age	int	11			用户年龄
phone	char	11			联系电话
address	varchar	64			联系地址
status	int	11		NOT NULL	信息状态
create_time	char	19		NOT NULL	创建时间
type	int	11		NOT NULL	用户身份
Update_time	char	19		NOT NULL	更新时间

表 4-3-2 社团表 (team)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
name	varchar	20		NOT NULL	社团名称
create_time	char	10		NOT NULL	创建时间
total	int	11		NOT NULL	社团人数
manager	char	13		NOT NULL	社团团长
type_id	char	13		NOT NULL	社团编号
Update_time	char	19		NOT NULL	更新时间
State	Varchar	255		NOT NULL	状态

表 4-3-3 社团类型表 (team_types)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
name	varchar	20		NOT NULL	类型名称
create_time	char	19		NOT NULL	创建时间
Update_time	char	19		NOT NULL	更新时间
State	Varchar	255		NOT NULL	状态

表 4-3-4 缴费表 (pay_logs)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
create_time	char	19		NOT NULL	缴费时间
total	double	0		NOT NULL	缴纳费用

表 4-3-5 通知表 (notices)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
title	varchar	20		NOT NULL	通知标题
detail	varchar	125		NOT NULL	通知详情
create_time	char	10		NOT NULL	创建时间
team_id	char	13			发布社团
State	Varchar	255		NOT NULL	状态

表 4-3-6 社团成员表 (members)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
create_time	varchar	20		NOT NULL	入团时间
team_id	varchar	125		NOT NULL	加入社团
user_id	char	10		NOT NULL	申请用户
State	Varchar	255		NOT NULL	状态

表 4-3-7 申请表 (apply_logs)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
status	int	11		NOT NULL	处理状态
create_time	char	19		NOT NULL	申请时间

表 4-3-8 活动表 (activities)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
name	varchar	20		NOT NULL	活动名称
comm	varchar	60		NOT NULL	活动概述
detail	varchar	256		NOT NULL	活动详情
ask	varchar	125		NOT NULL	活动要求
total	int	11		NOT NULL	报名人数
active_time	char	19		NOT NULL	活动时间
team_id	char	13		NOT NULL	发布社团
State	Varchar	255		NOT NULL	状态

表 4-3-9 活动日志 (active_logs)

字段名	字段类型	长度	主键或外键	字段值约束	对应中文属性名
id	char	13	主键	NOT NULL	记录 ID
create_time	char	19		NOT NULL	报名时间
active_id	char	13		NOT NULL	活动编号
user_id	char	13		NOT NULL	报名用户
Update_time	char	19		NOT NULL	更新时间

4. 典型功能加密设计

4.1. MD5加密设计

在本系统中，通过设计一个Encode方法，接受一个字符串 msg，通过 MessageDigest 获取 MD5 加密算法实例。随后，使用 digest() 方法对输入的字符串进行 MD5 加密，返回的字节数组经过Base64 编码后返回。而后通过MessageDigest.getInstance("md5")来获取 MD5 算法，MessageDigest 是 Java 中的消息摘要类，它可以对输入数据进行哈希处理。由于 MD5 输出的是

字节数组，因而直接输出字节数组可能不方便查看或传输，通常都会将其转换为可打印的Base64字符串编码。

实现场景：

在Index首页登录时，系统会记录一条日志，包含用户输入的用户名和密码，同时会通过userService查询数据库，看看用户输入的用户名是否存在。如果找到了对应的用户，就返回用户信息；如果没有找到，系统会返回null，并抛出错误信息“输入的用户名不存在”，然后停止登录过程。如果用户名存在，系统会对获取的密码进行加密处理，使用Md5Util.encode(password)方法将密码加密后，再和数据库中存储的加密密码进行比较。如果不匹配，系统会返回错误信息。验证密码通过后，系统会生成一个新的Token，将它和用户的ID一起缓存起来，并在响应中返回这个Token，之后的请求都可以通过这个Token来验证用户的身份。

4.2. UserID Base64加密

在用户登录后，需要在多个页面之间传递一些信息。例如，在展示用户的个人信息或创建新内容时，通常不会直接使用数据库中的用户ID，而是使用一种加密过的ID来传输数据。登录后，系统可能会生成一个加密的token（类似加密的用户ID）来管理会话，这样就可以用它来标识当前会话，而不暴露真实的用户ID。为了传输或存储这个加密后的ID，会先将用户ID转换成字符串，然后进行Base64编码。接着，把当前的时间戳也进行编码。然后，将编码后的用户ID和时间戳拼接在一起，再加上一部分时间戳来增加复杂性，避免被破解。最后，把拼接好的字符串字节数组反转，进一步增加破解难度。为了避免编码过程中出现的符号冲突，将字符串中的“=”符号替换成“#”，这样加密后的字符串既能传输用户信息，又能保护用户的隐私。