

Proj-UP!

课程项目组队系统

小组编号：204



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY



目录

- 1 项目背景、目的及意义
- 2 技术栈的选用
- 3 程序数据库部分的实现
- 4 程序前端和后端的实现
- 5 项目演示
- 6 项目总结与展望



1

研究背景、目的及意义

“在群里喊话不好吗”

- 课程项目组队的普遍需求

几乎每一所学校，每一个专业，都有需要进行项目组队的课程

- 现有解决方案

无法主动找到现在已经存在的队伍；未完全确定的队伍对新成员接收混乱；

在社交软件上盲目寻找队友效率低下；无法组队的同学的信息过于冗余；

任课教师难以对组队情况进行观察或干涉...

- 潜在的诉求

能不能利用数据库技术，获得一种 更好的组队信息获取方法？

项目组队的状况能否更公开、清晰？

任课教师有没有可能发挥监督者的作用？



需求分析回顾

别想了，没得选

2

-作为学生：

- 1.学生能在project内创建队伍并作为组长，可以为队伍提供队伍描述，可以邀请其它学生加入，也可以同意申请加入的学生加入，也可以将组员踢出队伍
- 2.学生能查看或检索project内未满人的队伍列表和队伍信息，并能申请加入队伍，申请时会提供申请内容和相关个人信息
- 3.每个队伍都有一个类似看板的board，组长在board上能添加project的进度，队员可以在里面留言或者在上面post重要的信息



-作为老师

- 1.能够为相应的课程创建project，为project提供信息，设置队伍最小人数、最大人数等参数和project时限，
- 2.在project下将学生邀请到课程相应的实验班让其自由组队。
- 3.能够直接将某个学生加入到某个队伍也可以直接将某个学生移出某个队伍
- 4.能够看到所有队伍的信息
- 5.能够对组队完毕后的project进行锁定
- 6.能够在project页面发announcement



数据库实现

来个人写一个小目录啥的

3

- 用户信息

id, 用户名, 密码, 用户身份

- 学生信息

学生id, 用户id, 学生个人信息(姓名, 性别, 班级, 年级, 住所等)

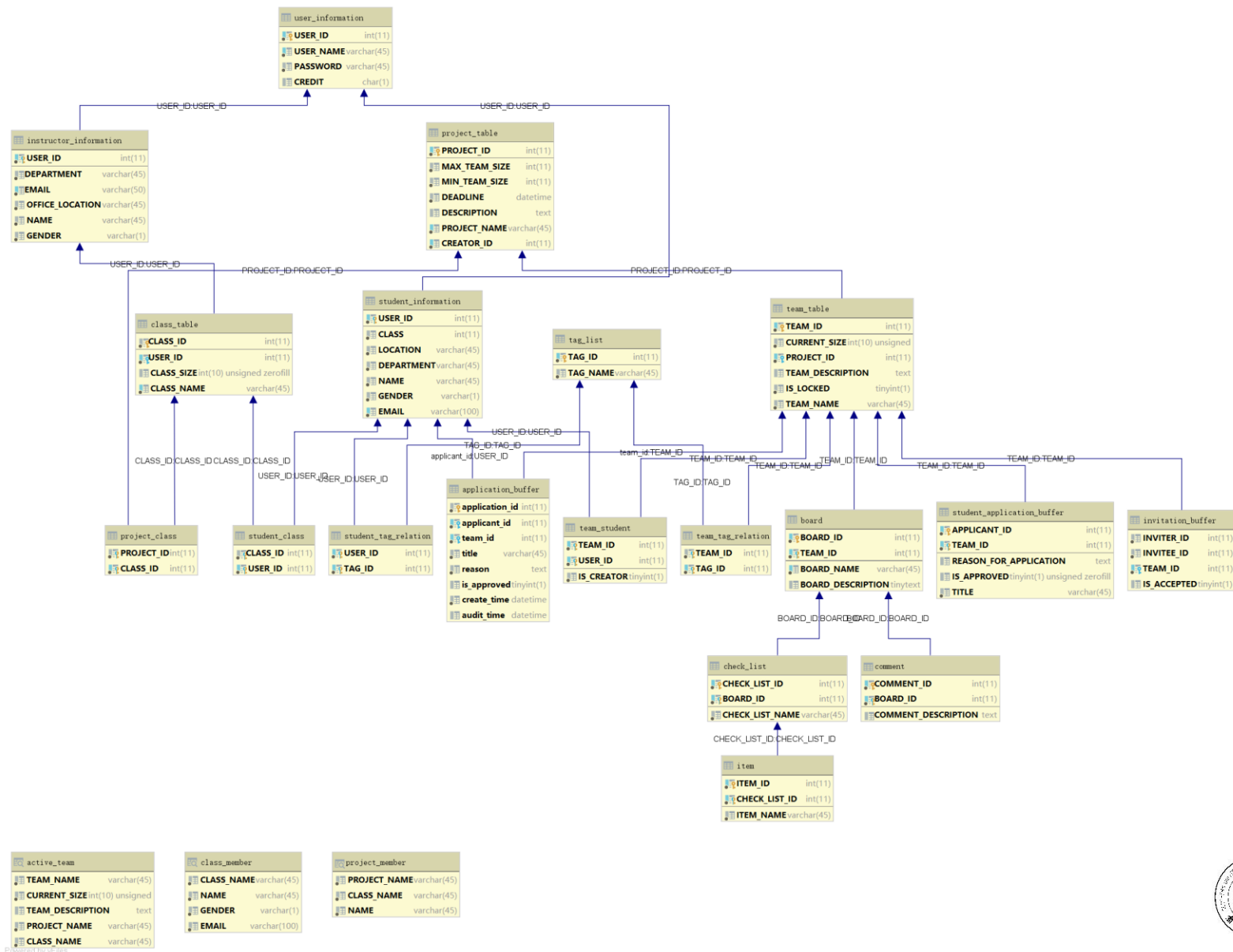
- 教师信息

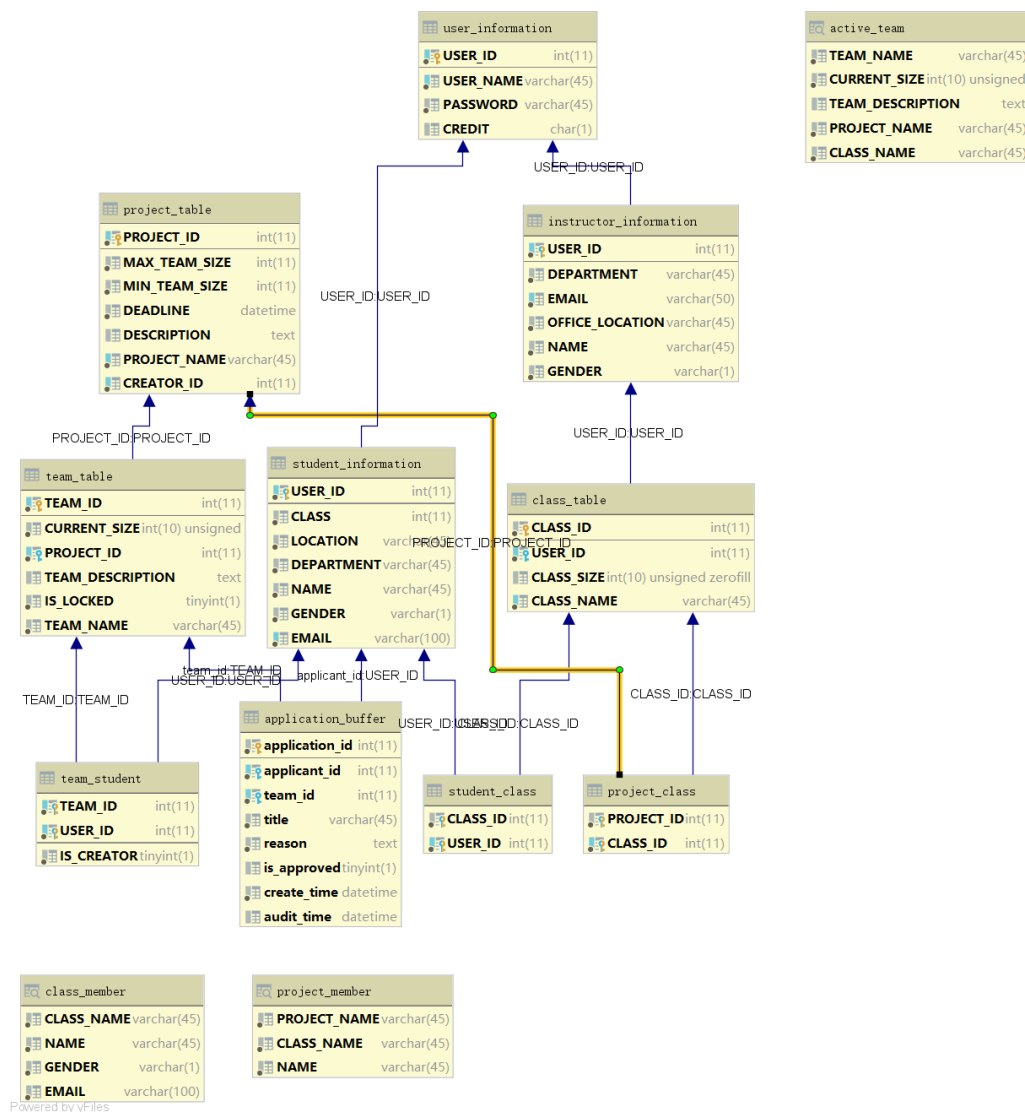
教师id, 用户id, 教室基本信息(姓名, 邮件, 电话)

- 项目信息

project id, creator id, 项目信息(名称, 组队人数设定, 描述, ddl...)

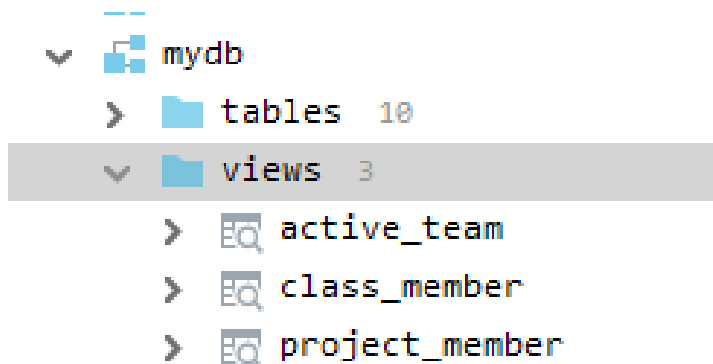






Powered by yFiles





active_team

TEAM_NAME	varchar(45)
CURRENT_SIZE	int(10) unsigned
TEAM_DESCRIPTION	text
PROJECT_NAME	varchar(45)
CLASS_NAME	varchar(45)

Powered by yFiles

class_member

CLASS_NAME	varchar(45)
NAME	varchar(45)
GENDER	varchar(1)
EMAIL	varchar(100)

project_member

PROJECT_NAME	varchar(45)
CLASS_NAME	varchar(45)
NAME	varchar(45)

mydb.active_team [10.21.93.251] × mydb.class_member [10.21.93.251] × mydb.application_buffer.update_isap

4 rows

<Filter criteria>

	TEAM_NAME	CURRENT_SIZE	PROJECT_NAME	CLASS_NAME
1	测试2队	1	cs307	lab1班
2	测试2队	1	cs307	lab2班
3	现世神圣	1	cs307	lab1班
4	现世神圣	1	cs307	lab2班



```
▼ tables 10
  ▼ application_buffer
    application_id int(11) (auto increment)
    applicant_id int(11)
    team_id int(11)
    title varchar(45)
    reason text
    is_approved tinyint(1)
    create_time datetime
    audit_time datetime
    PRIMARY (application_id)
    application_buffer_student_information_USER_ID_fk (applic
    application_buffer_team_table_TEAM_ID_fk (team_id) → team
    application_buffer_student_information_USER_ID_fk (applic
    application_buffer_team_table_TEAM_ID_fk (team_id)
    update_isApproved before update
```



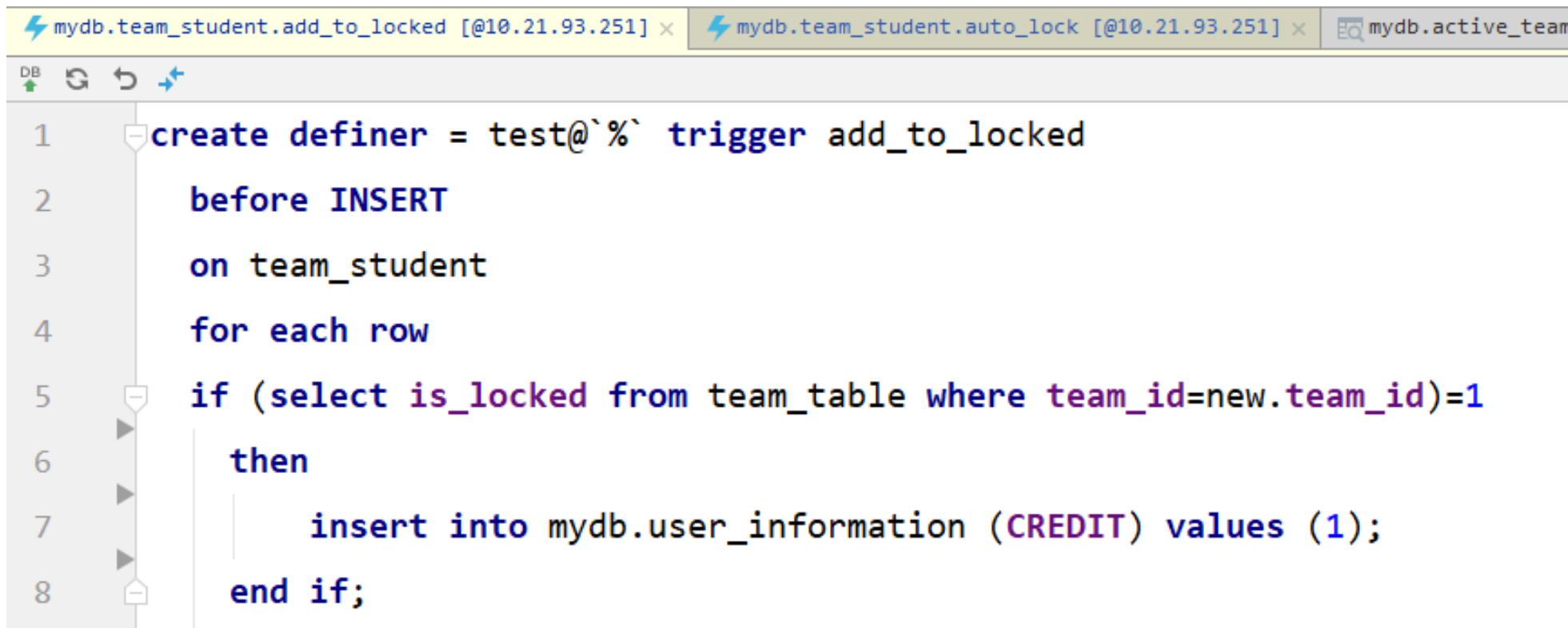
team_student

- TEAM_ID int(11)
- IS_CREATOR tinyint(1)
- USER_ID int(11)
- PRIMARY (TEAM_ID, USER_ID)
- fk_TEAM_MEBER_SHEET_STUDENT_INFORMATION1 (USER_ID) → student_information (USER_ID)
- FK_TEAM_MEMBER_SHEET_TEAM_TABLE (TEAM_ID) → team_table (TEAM_ID)
- fk_TEAM_MEBER_SHEET_STUDENT_INFORMATION1_idx (USER_ID)
- TEAM_ID_idx (TEAM_ID)
- add_teammate after insert
- add_to_locked before insert
- auto_lock after insert

team_table

- TEAM_ID int(11) (auto increment)
- CURRENT_SIZE int(10) unsigned
- PROJECT_ID int(11)
- TEAM_DESCRIPTION text
- IS_LOCKED tinyint(1)
- TEAM_NAME varchar(45)
- PRIMARY (TEAM_ID)
- TEAM_ID_UNIQUE (TEAM_ID)
- FK_TEAM_TABLE_PROJECT_TABLE (PROJECT_ID) → project_table (PROJECT_ID)
- PROJECT_ID_idx (PROJECT_ID)
- TEAM_ID_UNIQUE (TEAM_ID) UNIQUE
- delete_team before delete
- lock2notLock before update
- notLock2lock before update





The screenshot shows a MySQL IDE window with three tabs: 'mydb.team_student.add_to_locked [10.21.93.251] x', 'mydb.team_student.auto_lock [10.21.93.251] x', and 'mydb.active_team'. The active tab displays the following SQL code:

```
1 create definer = test@`%` trigger add_to_locked
2   before INSERT
3   on team_student
4   for each row
5   if (select is_locked from team_table where team_id=new.team_id)=1
6   then
7       insert into mydb.user_information (CREDIT) values (1);
8   end if;
```



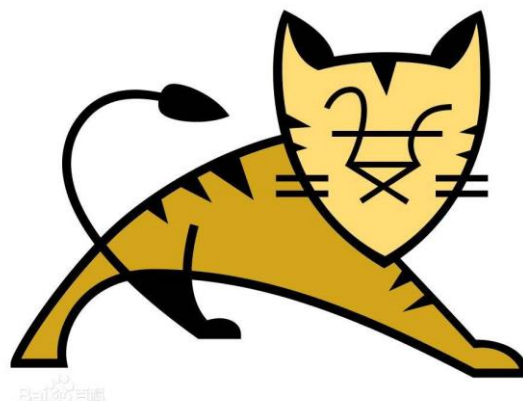
4

前端后端实现

同上，来个人写梗概



小程序



项目演示

视频时间!

5

项目总结与展望

感谢谷歌，感谢百度，感谢CSDN，感谢博客园，
感谢Stack Overflow，感谢模板作者1522姚瑶

6

- 项目总结与反思：

1. 解决了project组队难的问题；
2. 嵌入微信小程序便于推广；
3. 把太多SQL语句写在代码里；
4. 没有使用优秀的后端框架。





南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

感谢您的批评指正！

模板分享人：1522姚瑶
图片来源：南方科技大学官网、王本凡

备用演示



如果你看到了这一页，这说明视频炸了[头疼]