

“树协作”协作项目管理 微信小程序 [软件设计说明书]

目录

1. 引言	4
1.1 编写目的	4
1.2 命名规范	4
1.3 术语定义	4
1.4 参考资料	4
1.5 相关文档	4
1.6 版本更新信息	4
2. 总体设计	4
2.1 硬件运行环境	4
2.2 软件运行环境	4
2.3 子系统清单	5
2.4 功能模块清单	5
3. 数据库设计	6
3.1 数据库中表名列表	6
3.2 数据库中表的关系	6
3.3 数据库表的详细清单	7
4. 典型功能子系统设计	9
1. 用户信息管理系统设计	9
2. 消息提醒系统设计	14
3. 发现系统设计	16
4. 项目管理系统设计	19
5. 信用评价系统设计	23
5. 功能模块设计	25
1. 主界面 home.wxml	25
2. 个人中心 personal_center.wxml	25
3. 修改个人信息页面 info_change.wxml	25
4. 新建项目页面 new_project.wxml	26
5. 项目页面 project.wxml	26
6. 发现页面 search.wxml	27
7. 消息中心页面 info_center.wxml	27

8. 项目结算页面 project_settle.wxml	27
6. 存储过程设计	28
1. getSuperiors()	28
2. getSubordinate()	28
7. 接口设计	29
7.1 用户接口	29
7.2 外部接口	29
7.3 内部接口	29
8. 角色授权设计	29
9. 系统错误处理	29
9.1 出错信息	29
9.2 故障预防与补救	30
9.3 系统维护设计	30
10. 测试计划	30

1. 引言

1.1 编写目的

在完成“树协作”协作项目管理小程序系统需求分析的基础上，我们依据《需求规划说明书》对该系统进行了设计。为了文档化对系统的设计，编写本文档。

1.2 命名规范

本文档中所称“小程序”，约定皆为微信小程序。

数据库表面、字段名用大写开头的英文命名。

私有变量用“_”开头的小写英文命名。其他的变量用大写英文字母开头。

所有的函数用大写英文字母开头。

1.3 术语定义

总体结构：软件系统的总体逻辑结构，本系统采用面向对象的方法对系统进行设计。

概念数据模型(CDM)：关系数据库的概念设计模型，主要表现为 E-R 图。

物理数据模型(PDM)：关系数据库的物理设计模型。

1.4 参考资料

[1]吕云翔.软件工程实用教程.北京：清华大学出版社，2015.

1.5 相关文档

《需求规划说明书》

1.6 版本更新信息

版本更新记录如下表

版本号	创建者	创建日期	维护者	维护日期	维护纪要
V1.0	袁昊	2019-5-1	袁昊	2019-5-13	
V1.1	袁昊	2019-5-24	袁昊	2019-5-26	

2. 总体设计

2.1 硬件运行环境

- (1) 0.8GHz 及以上频率单或多核 CPU
- (2) 1G 及以上 ROM 存储空间
- (3) 512M 及以上 RAM 内存空间

2.2 软件运行环境

满足其下任一即可

- (1) 安卓手机：安卓 4.0 以上系统适用屏幕像素。

- (2) iPhone: iPhone6.1-iPhone8.0 以上系统适用屏幕像素
 (3) iPad: iPad6.1-iPad8.0 以上系统适用屏幕像素
 (4) Windows Phone8 版手机: Windows Phone8.0 以上系统适用屏幕像素

2.3 子系统清单

子系统清单见下表:

子系统编号	子系统名称	子系统功能描述
1	用户信息管理系统	(1) 注册、登录、注销 (2) 个人信息补全、修改 (3) 管理员对用户的操作
2	项目发起系统	(1) 发起项目: 填写项目信息、需求等 (2) 在项目下发起子项目
3	项目追踪管理系统	(1) 浏览项目进度、提交项目 (2) 修改、关闭项目 (3) 树状形式浏览项目
4	项目结算系统	(1) 根节点所有子项目完成后可结算 (2) 可选根据成员排名综合计算贡献度
5	历史查看系统	查看历史参与项目与状态
6	信用评价系统	(1) 根据历史项目参与情况计算信用度 (2) 提供根据信用度情况进行筛选、分级
7	消息提醒系统	(1) 在主界面显示、提醒参与项目 (2) 接近 DDL 时对参与者与组长进行提醒
8	发现系统	搜寻并加入公开项目

2.4 功能模块清单

功能清单见下表:

模块编号	名称	模块功能描述
101	登录	使用微信号登录进小程序
102	查看个人信息	注册用户或管理员浏览个人信息
103	修改个人信息	注册用户或管理员对个人信息进行修改
201	项目发布	注册用户填写信息、发布项目
202	子项目发布	注册用户在项目下自创新的子项目
301	修改项目	项目发起人修改项目信息
302	浏览项目	项目成员或管理员浏览项目信息以及进度
303	项目工作提交	项目成员提交项目工作
304	项目提交	项目发起人提交子项目, 同时关闭子项目
305	关闭项目	项目发起人或管理员强制关闭项目
306	邀请加入	发送链接, 邀请注册或非注册用户加入项目
401	项目结算	项目发起人提交根项目、进入结算状态, 通过互评结算所有子项目的贡献度, 得出所有子项目贡献度后关闭项目
402	成员互评	项目中每位成员对所有同项目成员进行排名, 最终综合

		计算得出贡献度，返回到结算中
501	查看历史项目	用户查看自己参与过的已关闭项目信息
601	信用度计算	每当一个项目关闭，通过参与者完成状态反馈到个人信用中
701	主界面项目列表	在主界面显示目前参与的未关闭项目
702	DDL 提醒	在临近项目 DDL 三天内时，对参与者与发起者进行提醒
801	发现项目	注册用户在“发现”菜单中浏览可参与公开项目以及其详细信息
802	加入发现项目	注册用户在“发现”菜单中申请参与公开项目

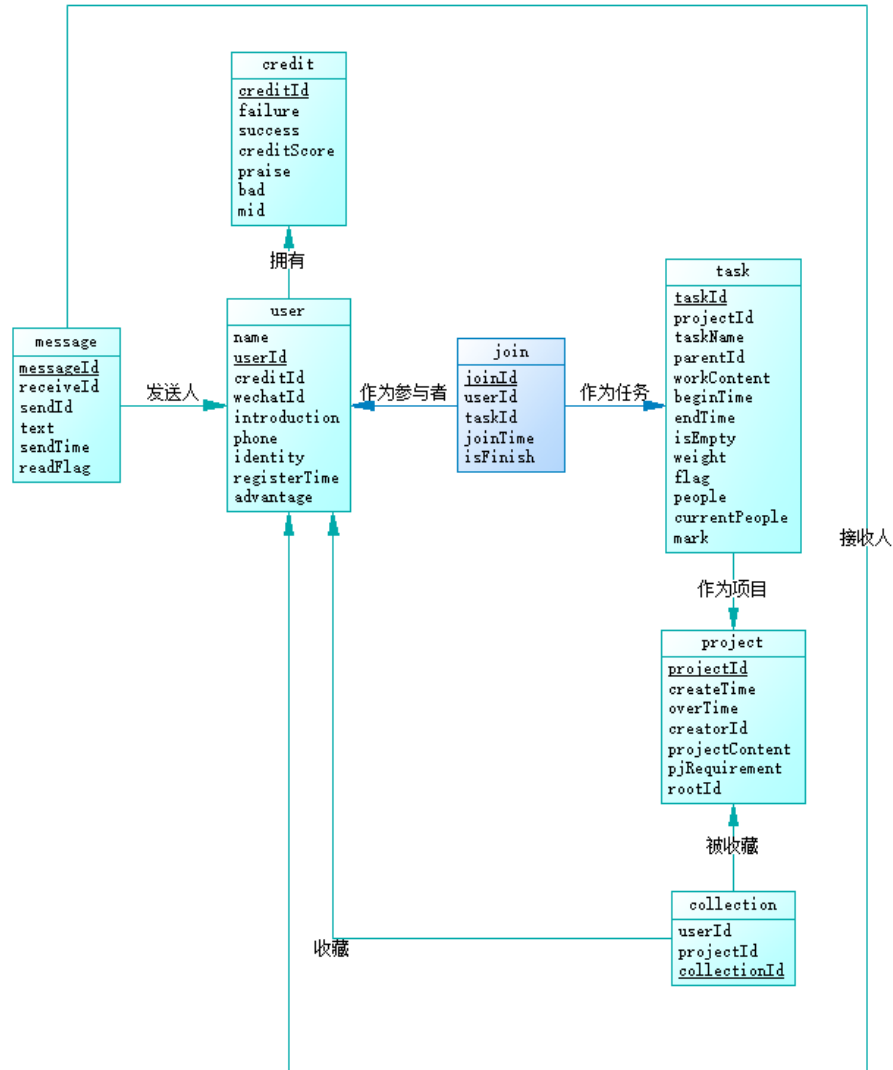
3. 数据库设计

3.1 数据库中表名列表

编号	表名	表功能说明
1	user	用户表
2	project	项目表
3	message	信息表
4	task	工作任务表
5	collection	用户收藏项目表
6	credit	用户信用积分表
7	join	加入表

3.2 数据库中表的关系

表中关系见图



3.3 数据库表的详细清单

用户表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
用户名	name	char(15)	15				X
用户 ID 号	userId	int		X	X		X
信用 ID 号	creditId	int				X	X
微信标识号	wechatId	char(30)	30				X
介绍	introduction	char(250)	250				
联系手机	phone	char(15)	15				
是否通过验证	identity	bool					X
注册时间	registerTime	date					X
优势	advantage	char(20)					

参与表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
参与 ID	joinId	int		X	X		X
参与者 ID	userId	int				X	
任务 ID	taskId	int				X	
加入时间	joinTime	datetime					X
完成标记	isFinish	boolean					X

项目表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
项目 ID	projectId	int		X	X		X
项目名	projectName	char(20)	20				X
创建时间	creatTime	datetime					X
结束时间	overTime	datetime					X
创建者 id	creatorId	int					X
项目内容	projectContent	char(100)	100				X
项目要求	projectRequire	char(100)	100				

工作任务表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
任务 id	taskId	int		X	X		X
项目 id	projectId	int				X	X
工作名	taskName	char(20)	20				X
父结点 id	parentId	int					X
主要内容	workContent	char(100)	100				
开始时间	beginTime	datetime					X
结束时间	endTime	datetime					X
空标记	isEmpty	bool					X
权重	weight	float(8)	8				X
项目类型标记	projectflag	int					X
当前人数	currentPeople	int					X
招募人数	needPeople	int					X
项目完成标记	finishFlag	int					X

收藏表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
收藏 id	collectionId	int		X	X		X
收藏者 id	userId	int				X	X
项目 id	projectId	int				X	X

信息表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
信息 ID	messageId	int		X	X		X
接收者 ID	receiverId	int				X	X
发送者 ID	senderId	int				X	X
内容	text	datetime					X
发送时间	sendTime	datetime					X
已读标志	readFlag	bool					X

信用表如图

中文列名	代码名	数据类型	数据长度	自增	主键	外键	不能为空
信用 id	creditId	int		X	X		X
失败次数	failure	int					
成功次数	success	int					
信用分	creditScore	int					
好评	praise	int					
差评	bad	int					

4. 典型功能子系统设计

1. 用户信息管理系统设计

类设计

该功能系统用于实现用户的注册，登录和注销，个人信息补全，修改以及管理员对用户的相关操作，类图设计如图 1 所示

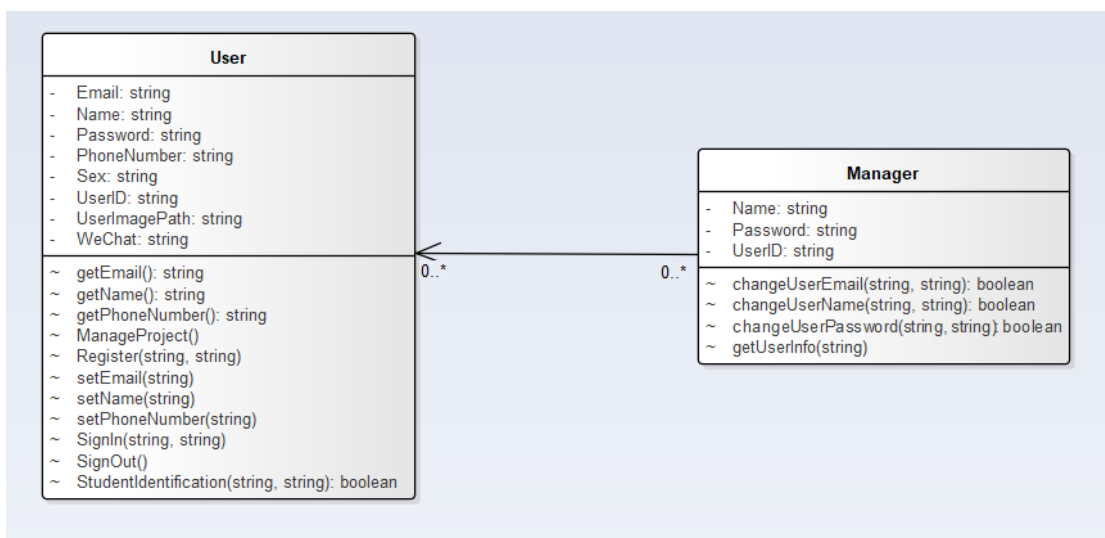


图 1 用户信息管理系统类图

活动图设计

以用户信息管理系统中修改用户信息为例，当用户发出修改信息请求后，系统跳转至修改信息界面，用户发出确认请求后，数据库更新用户信息，更新成功则修改完成，返回到用户信息界面，更新失败则给出相应的错误信息。该功能模块活动图如图 2 所示。

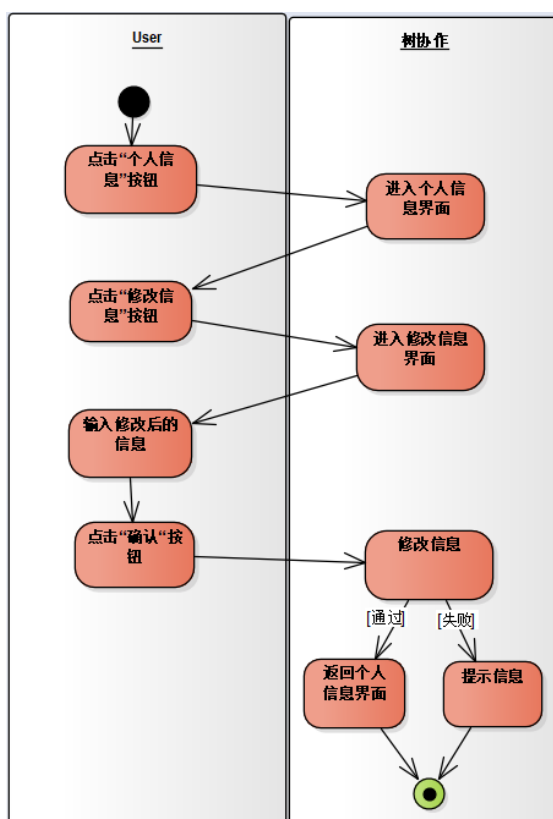


图 2 修改信息活动图

时序图设计

图 3, 4, 5, 6, 7, 8 分别描述了用户注册，登录，注销，查看个人信息，修改个人信息，校园信息认证的执行序列过程。

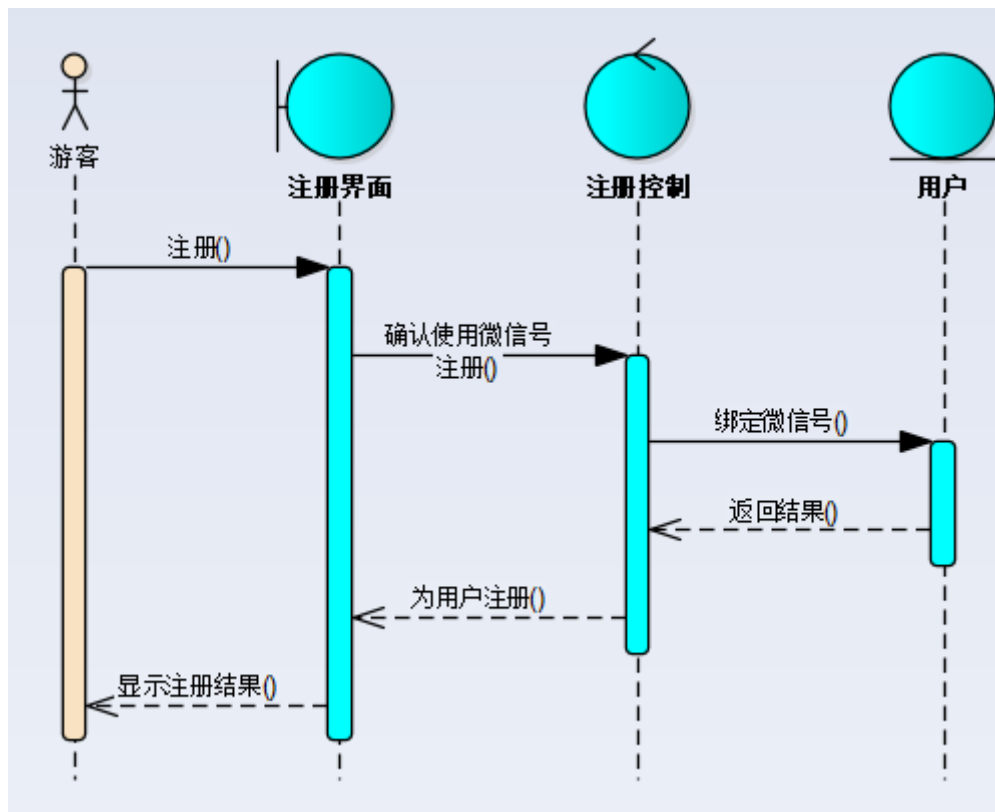


图 3 注册序列图

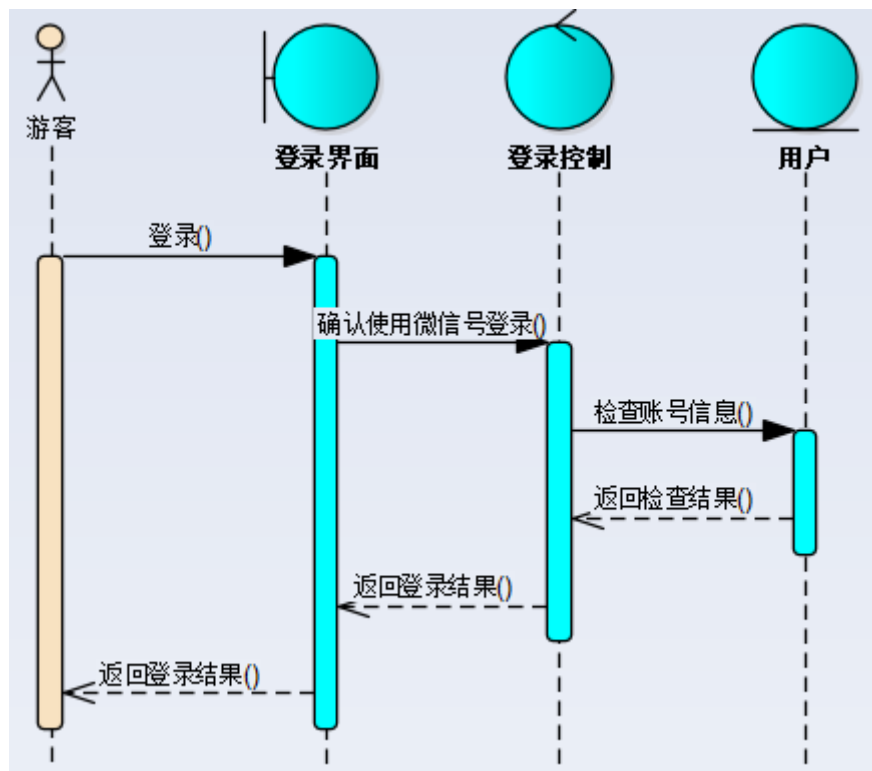


图 4 登录序列图

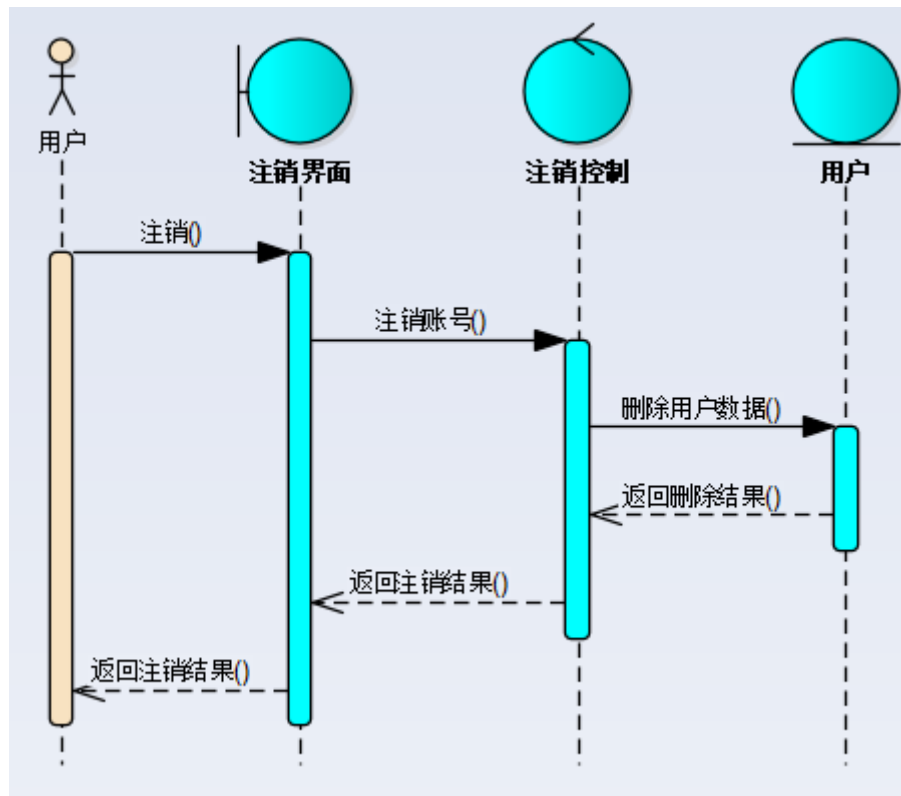


图 5 注销序列图

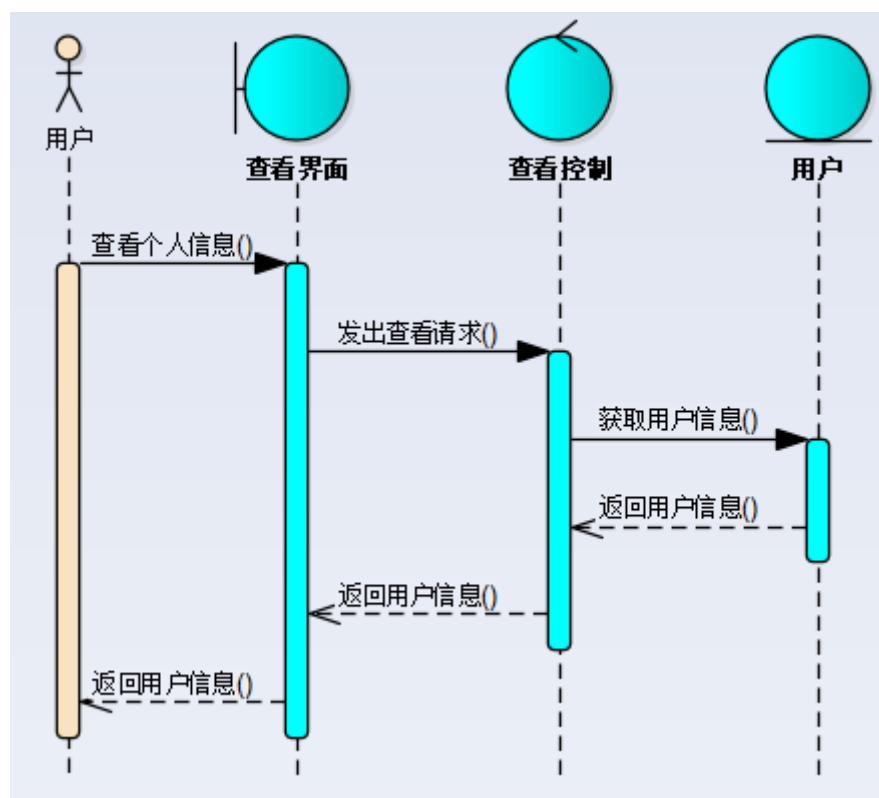


图 6 查看个人信息序列图

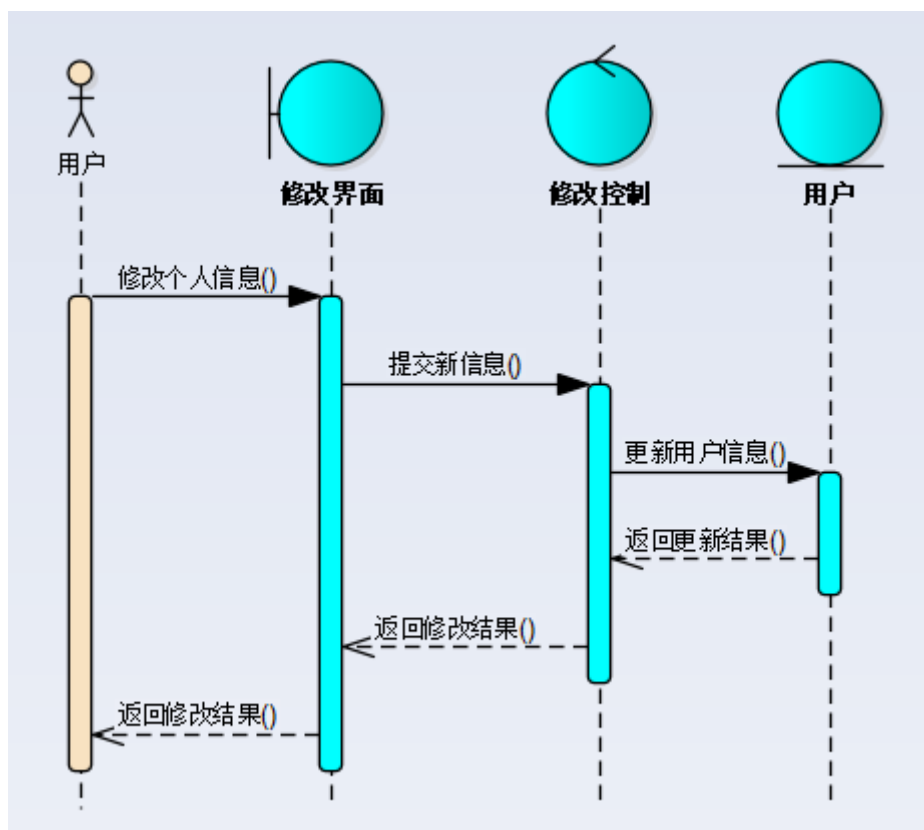


图 7 修改个人信息序列图

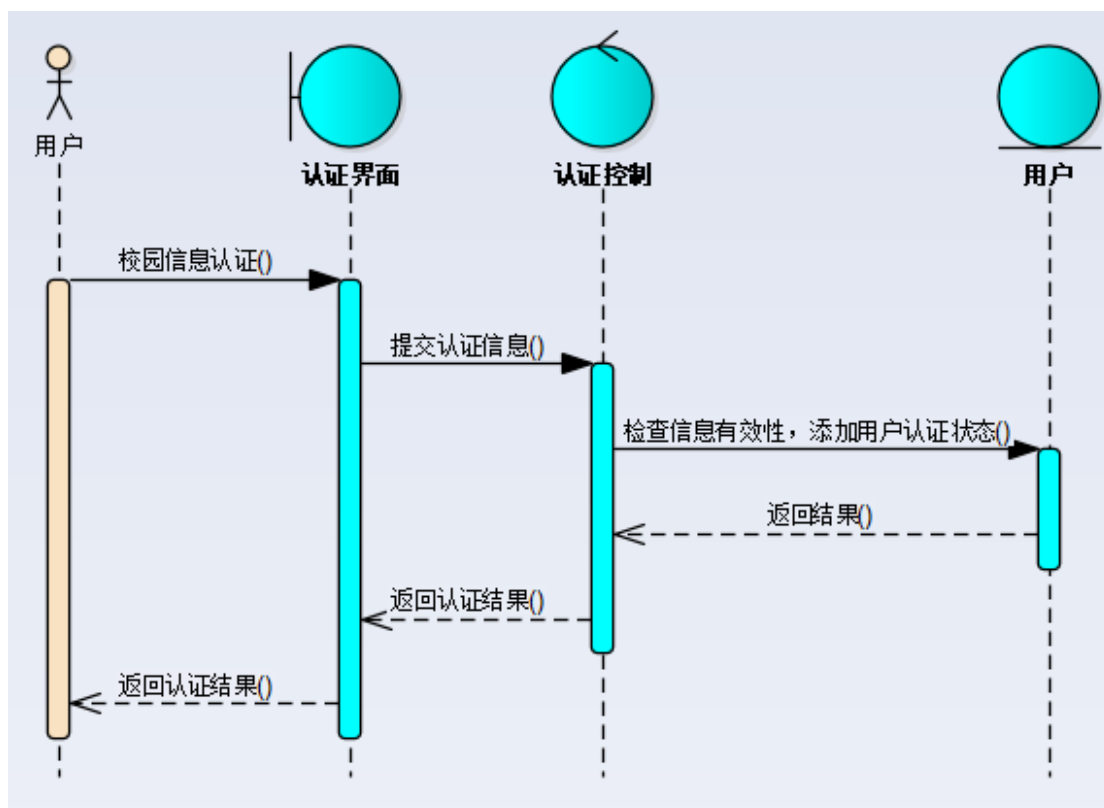


图 8 校园信息认证序列图

2. 消息提醒系统设计

类设计

该功能系统用于实现用户的收取信息，系统对用户发送信息，类图设计如图 1 所示

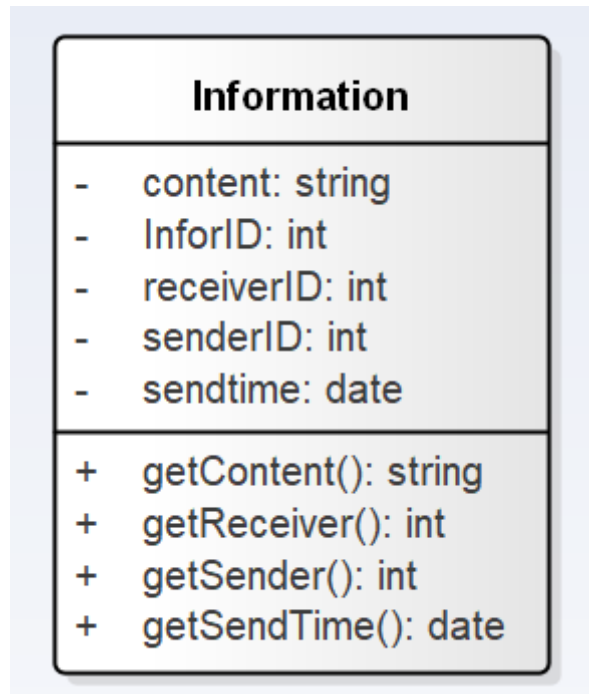


图 1 用户消息提醒系统类图

活动图设计

以用户消息提醒系统中用户给用户发送消息为例，当用户发出发送消息请求后，系统获取消息，并从数据库中检索收取消息的用户的信息，并将消息存放于对应用户的消息列表中。该功能模块活动图如图 2 所示。

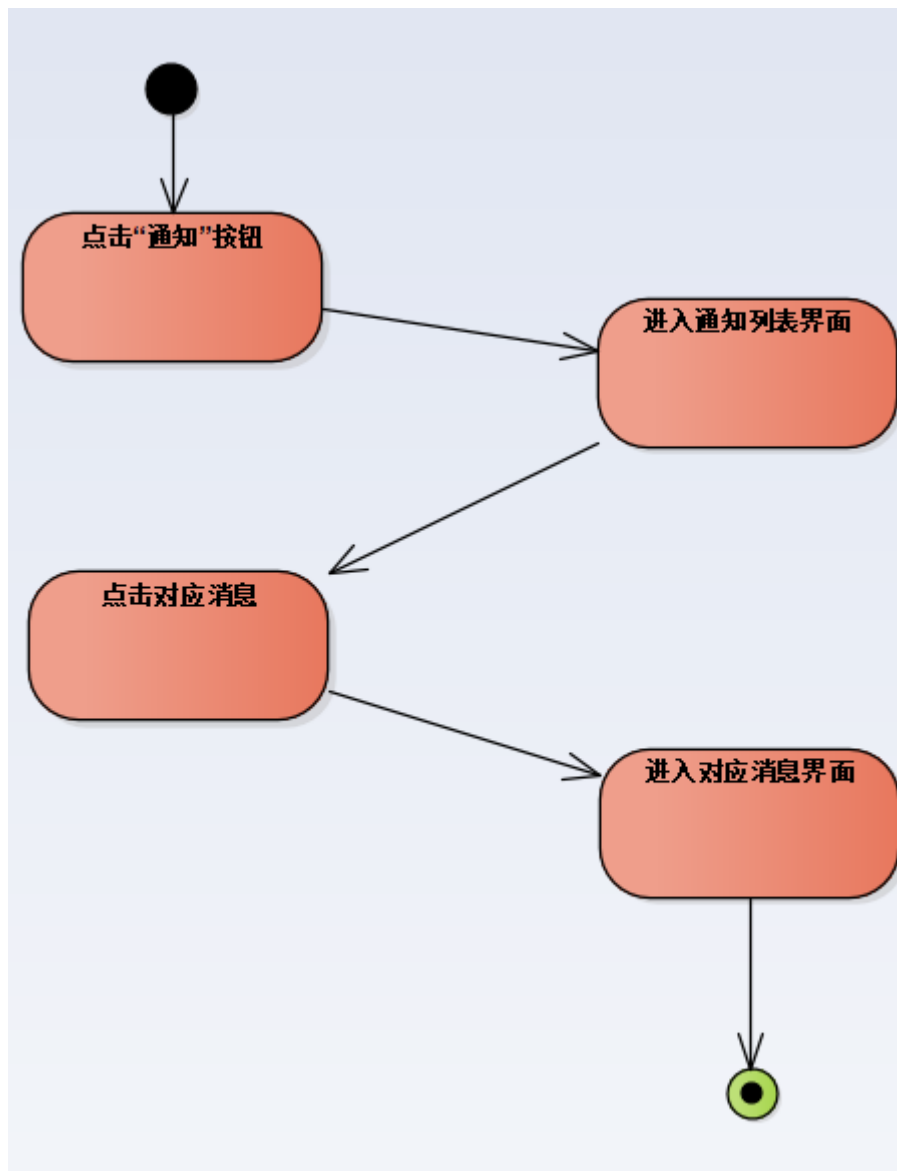


图 2 查看通知消息活动图

时序图设计

图 3 分别描述了用户给用户发送消息的执行序列过程。

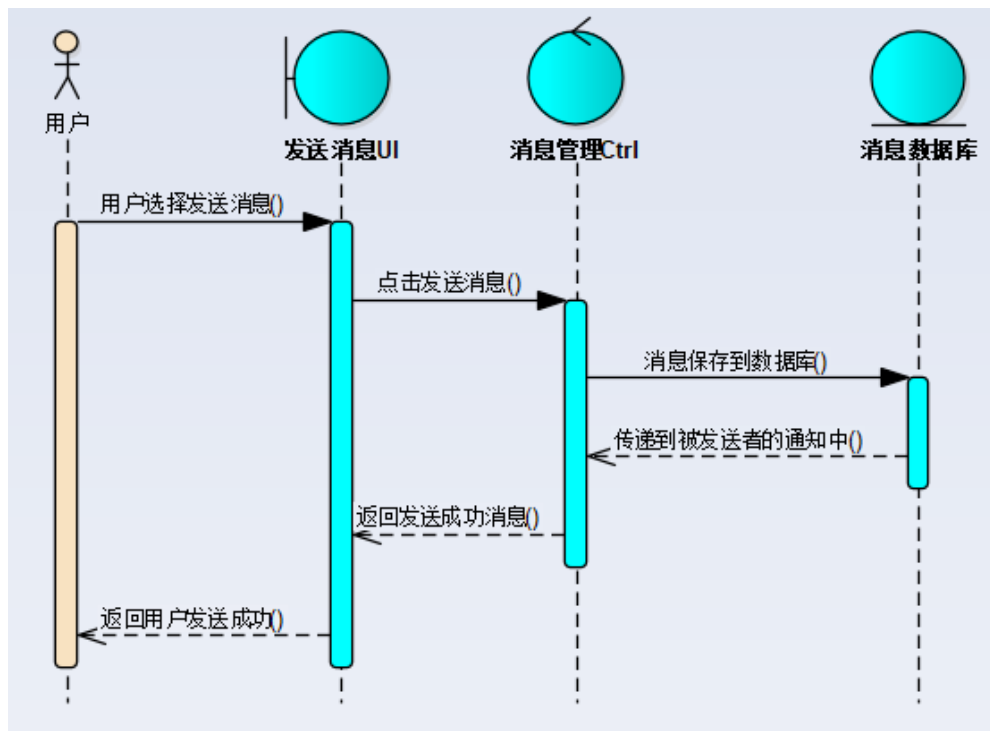


图 3 用户给用户发送消息序列图

3. 发现系统设计

类设计

该功能系统用于实现用户的根据关键词搜索，不根据关键词，加入活动。类图设计如图 1 所示

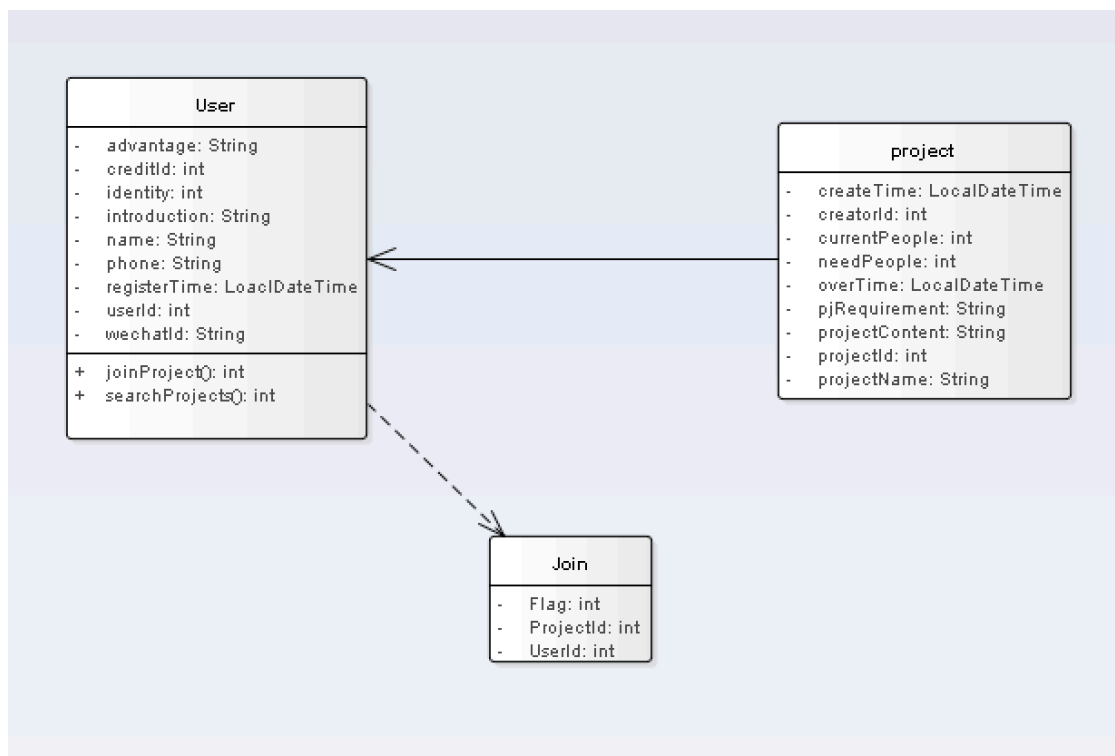


图 1 用户信息管理系统类图

活动图设计

以搜索活动信息为例，当用户输入关键词或者不输入关键词 后，系统检索活动信息，并更新发现界面，该功能模块活动图如图 2 所示。

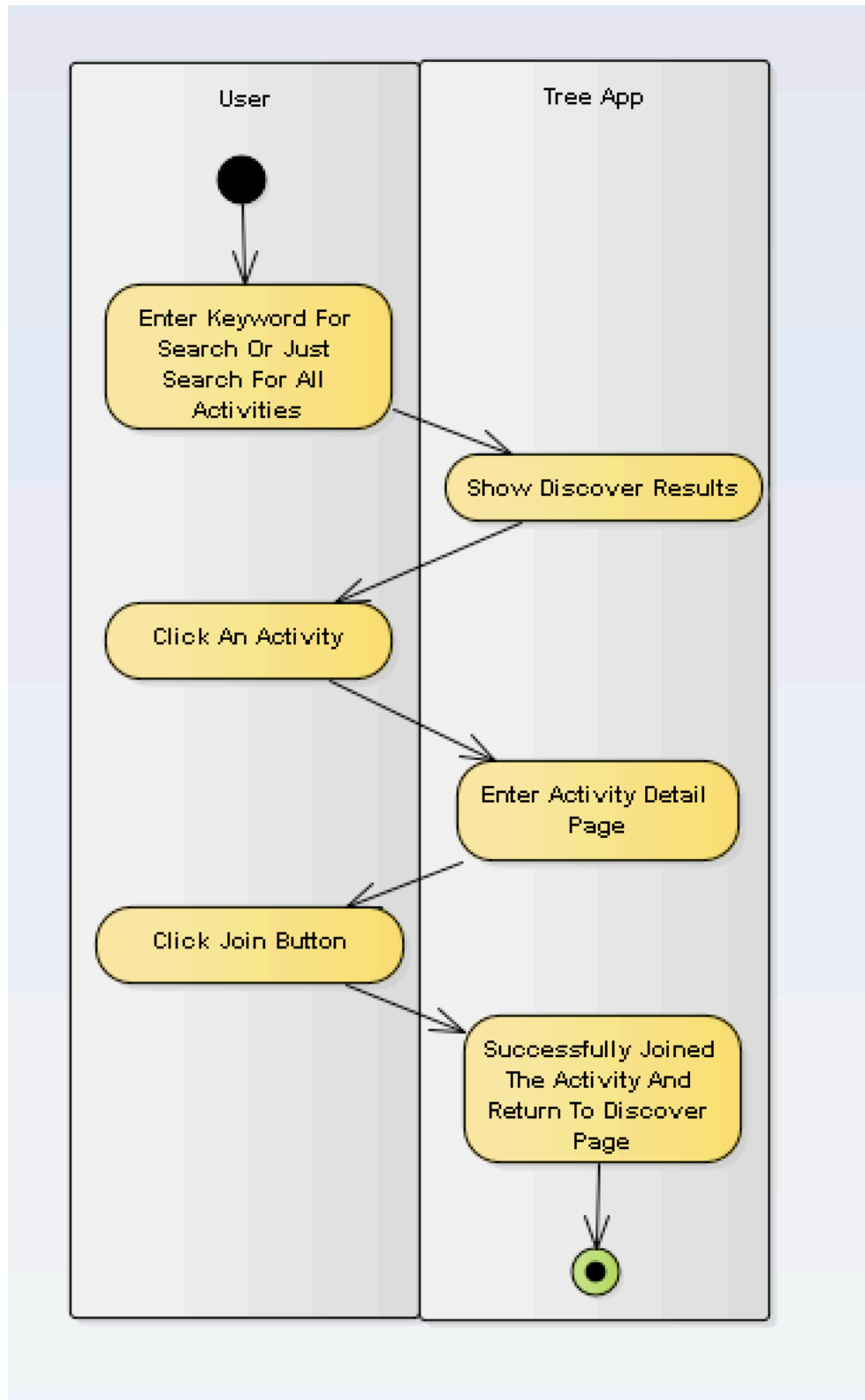


图 2 搜索活动活动图

时序图设计

图 3, 4 分别描述了用户搜索活动, 加入活动 的执行序列过程。

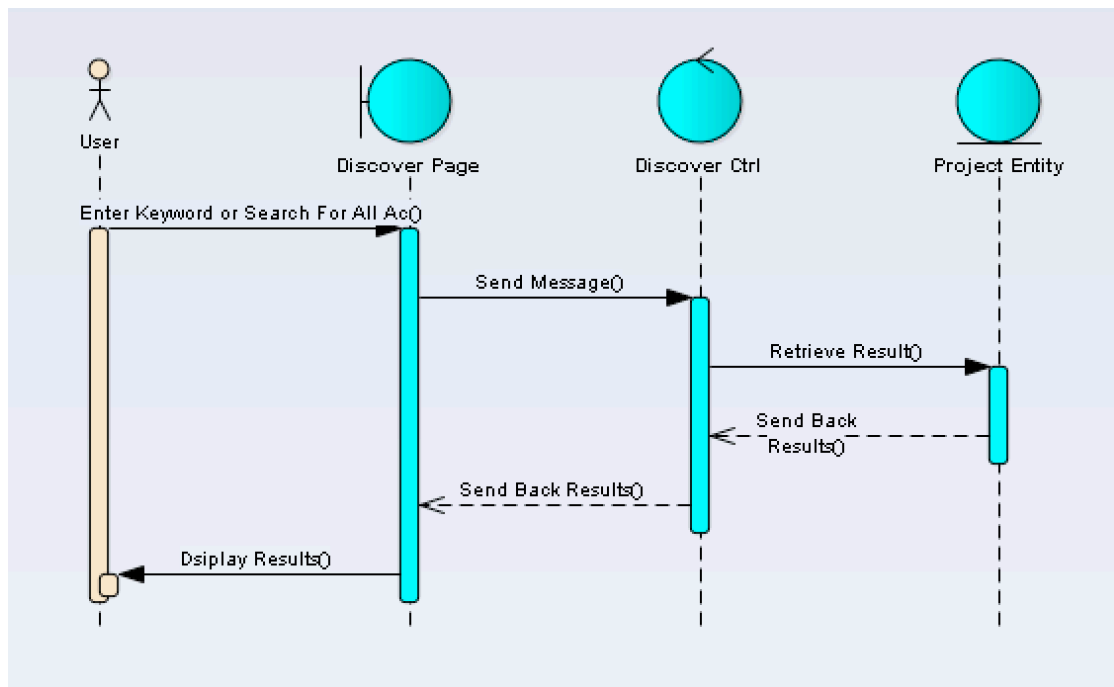


图 3 搜索活动图

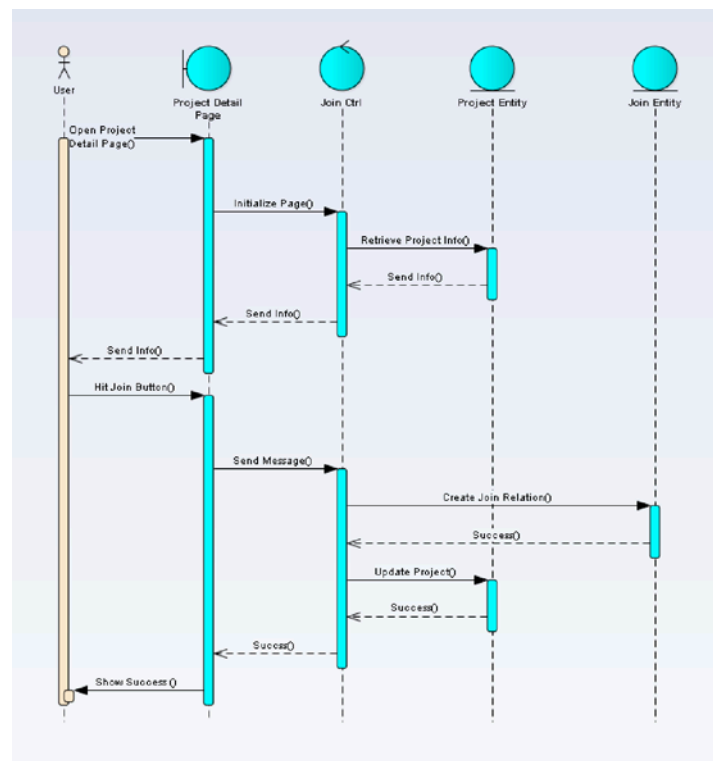
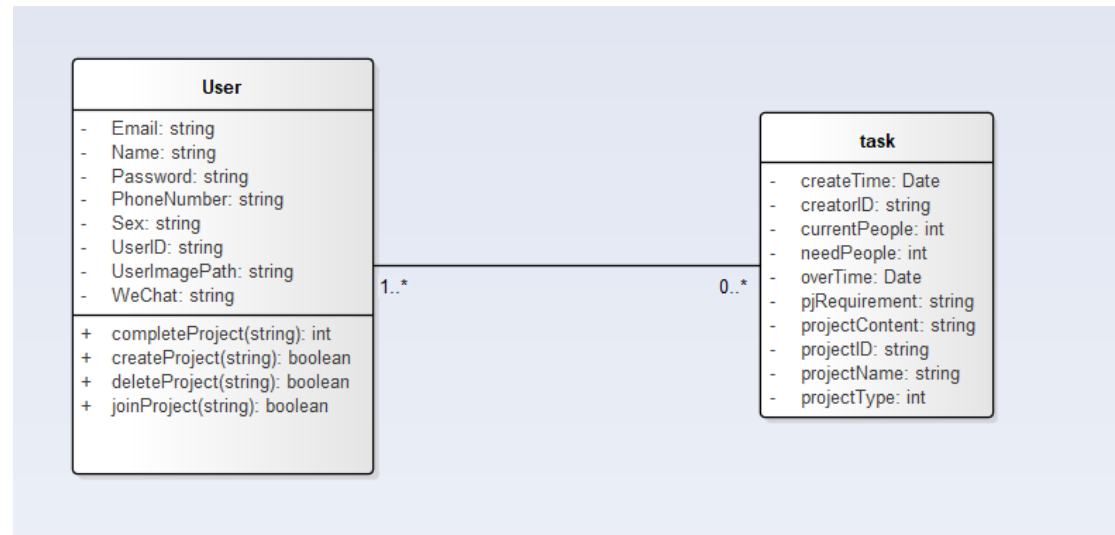


图 4 加入活动序列图

4. 项目管理系统设计

类设计

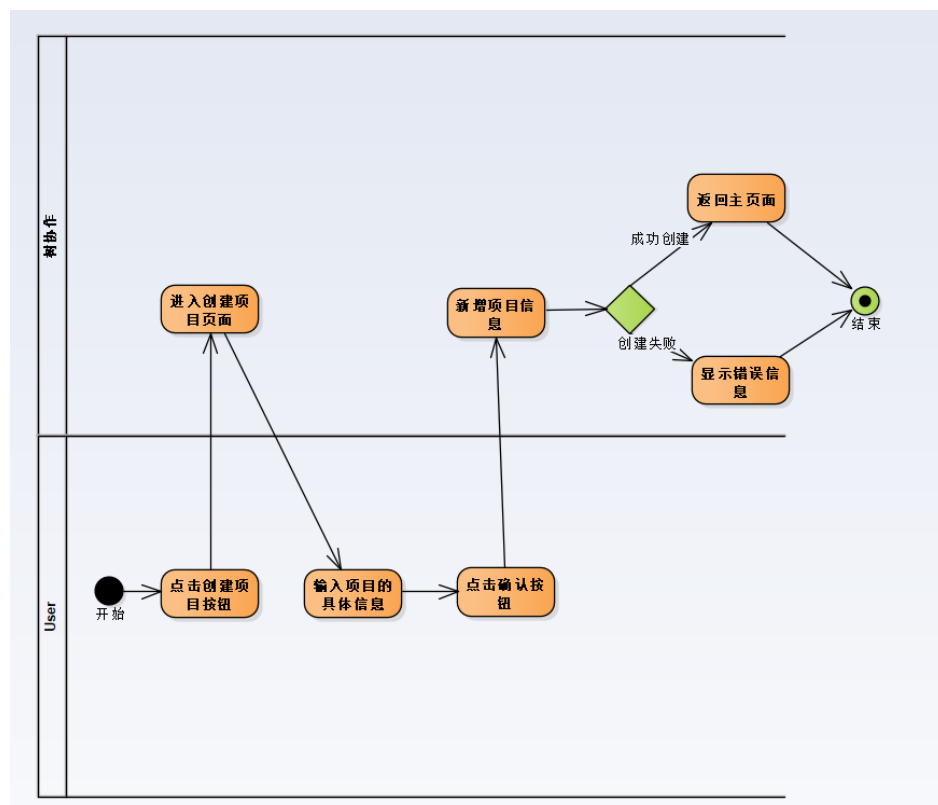
该功能系统用于实现用户对项目的创建、删除、加入和完成类图设计如下图所示



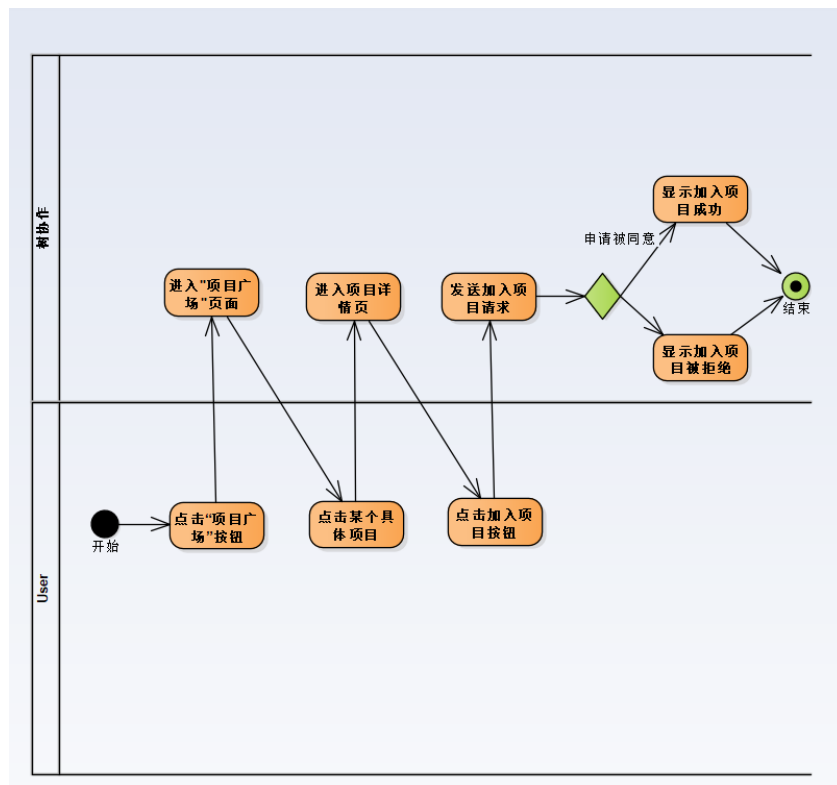
活动图设计

用户创建项目：

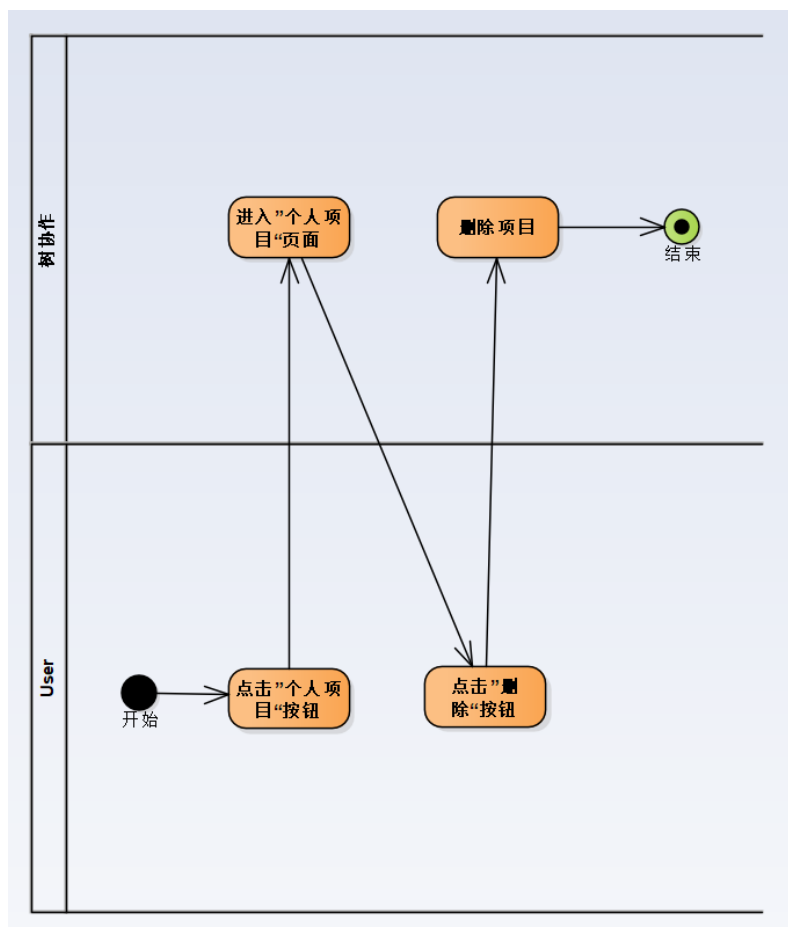
用户可以在主页面直接点击创建项目的按钮跳转进入项目创建页面，在该页面填写项目信息后可以提交项目创建申请，经审核后告知用户是否成功创建项目。



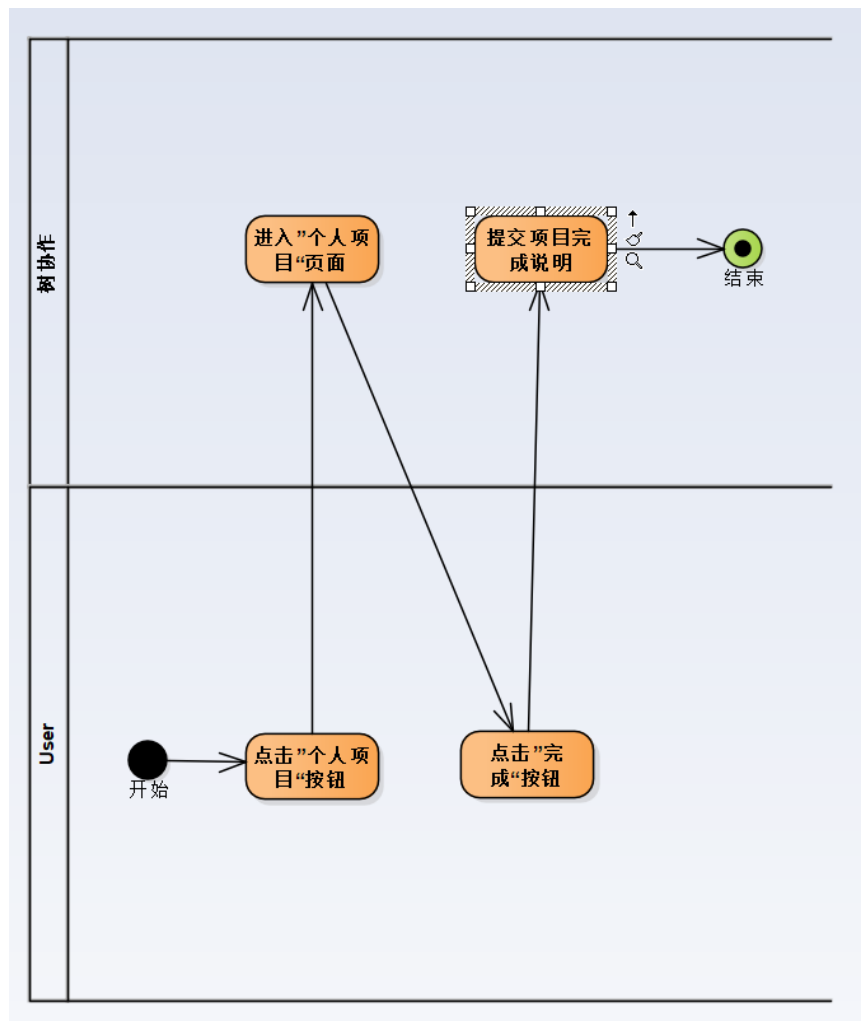
用户加入项目：



用户删除项目：

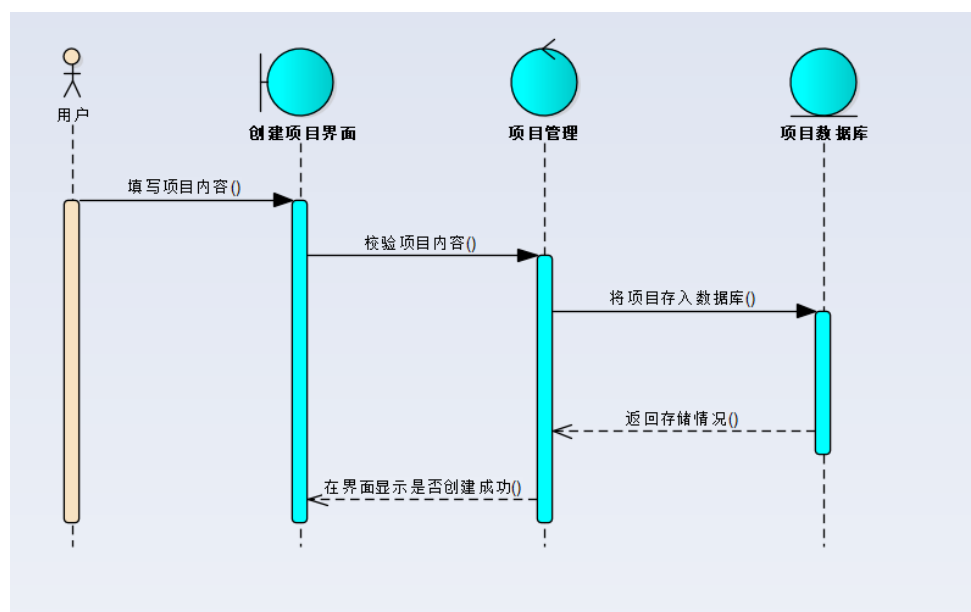


用户完成项目：

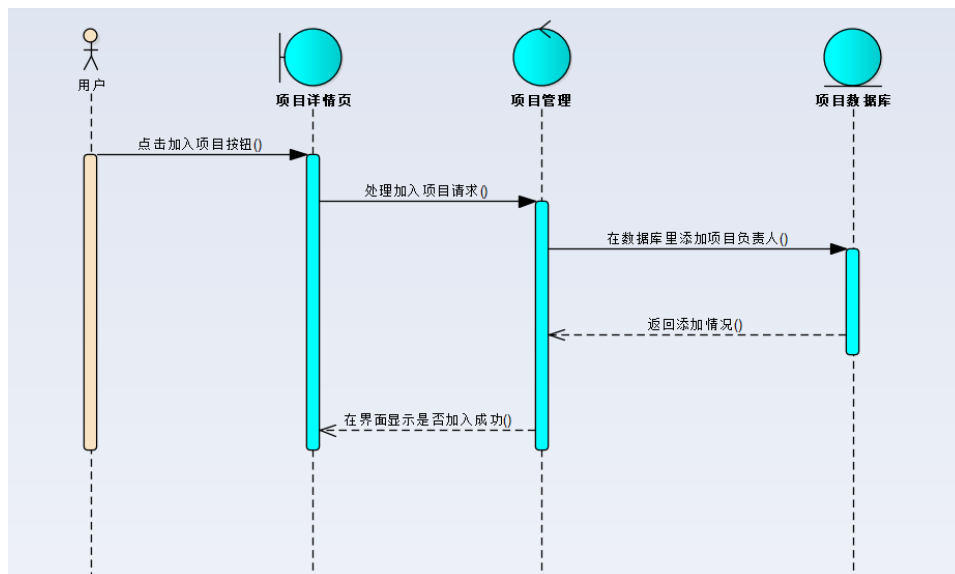


时序图设计

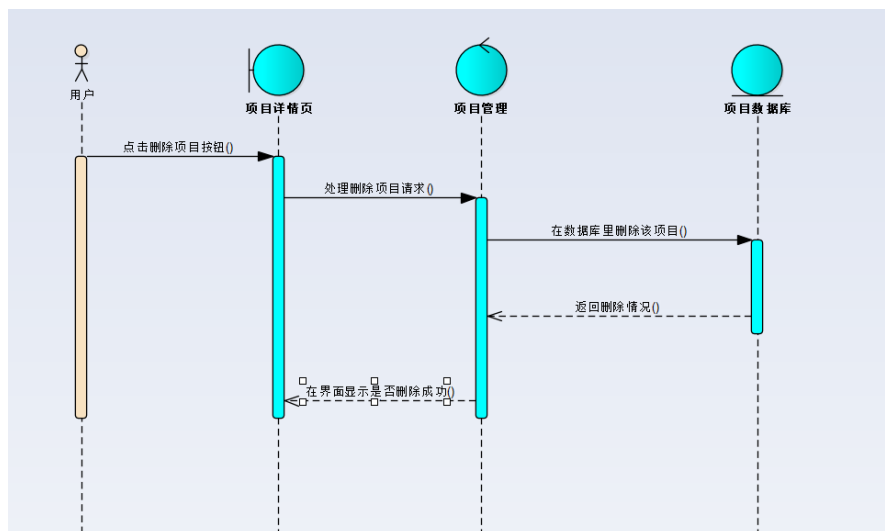
用户创建项目时序图：



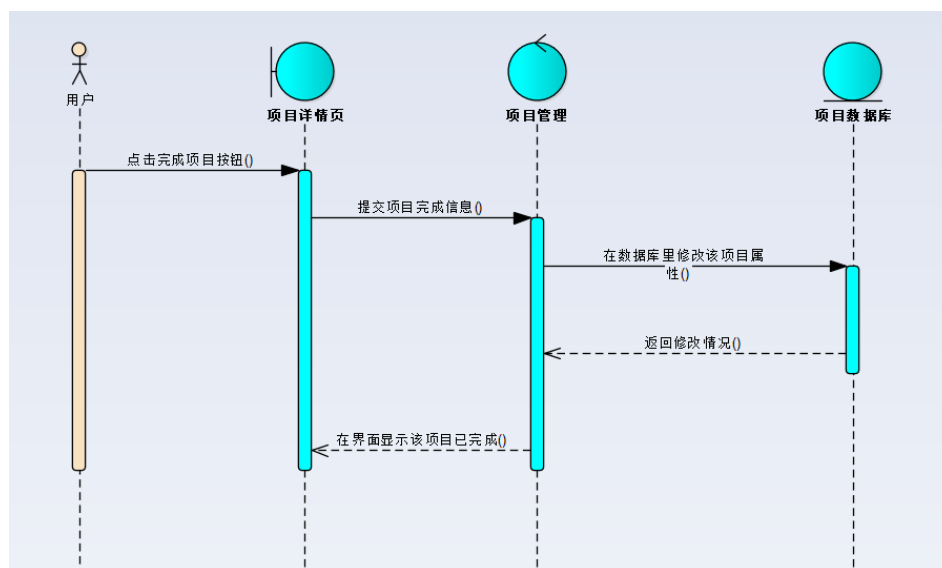
用户加入项目时序图：



用户删除项目：



用户完成项目：



5. 信用评价系统设计

类设计

该功能系统用于实现信用评价的相关操作，类图设计如图 1 所示

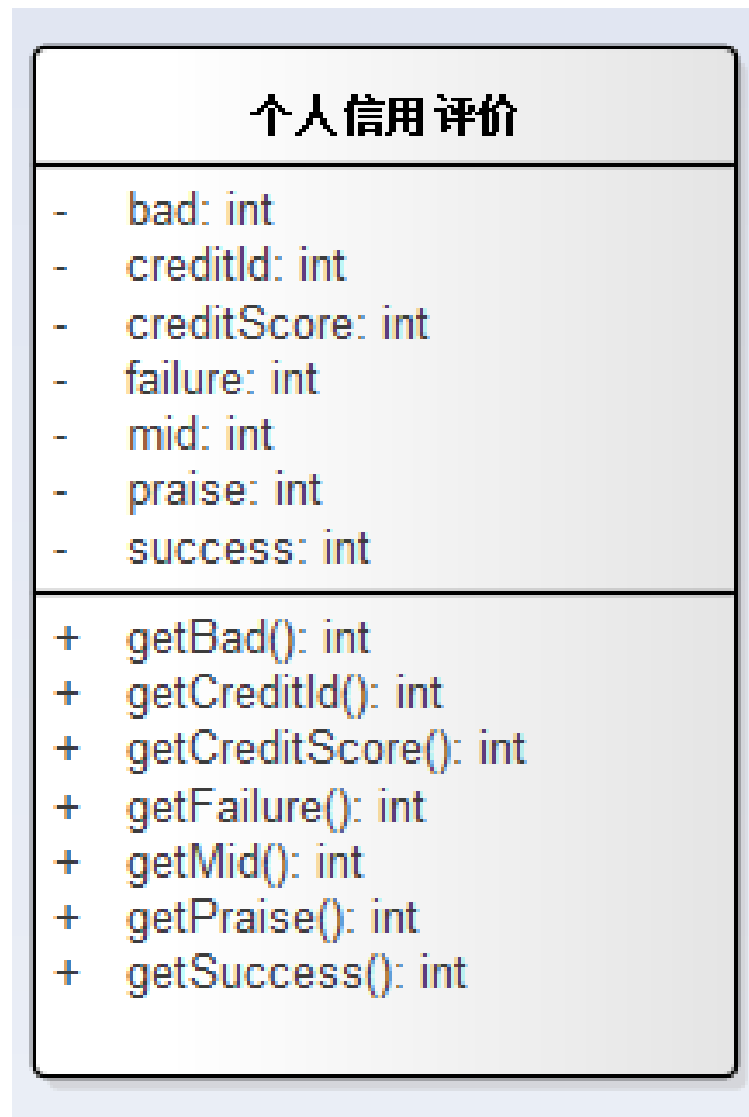


图 1 用户信用评价系统类图

活动图设计

以用户信用评价系统中评价信用为例，当用户发出评价信用请求后，系统跳转至评价信用界面，用户发出确认评价信息请求后，数据库更新用户评价信息，更新成功则修改完成，返回到用户信息界面，更新失败则给出相应的错误信息。该功能模块活动图如图 2 所示。

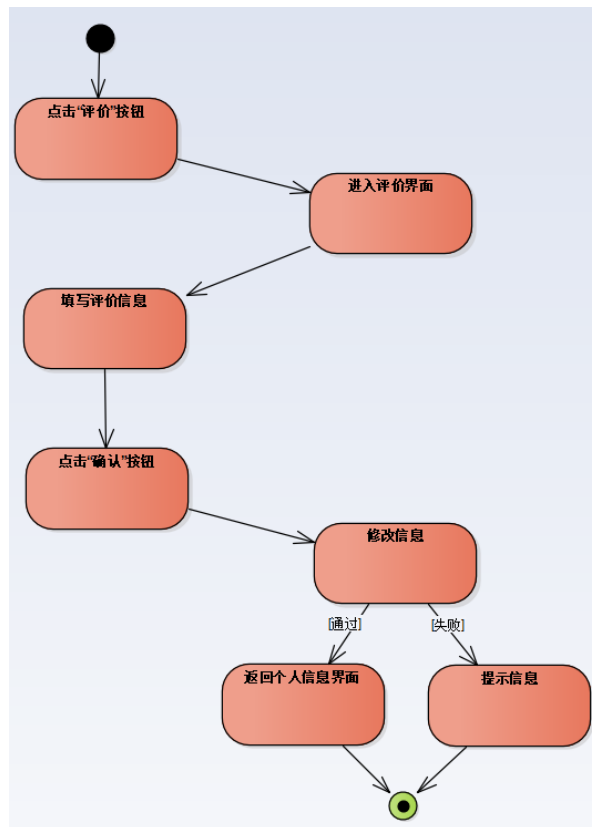


图 2 信用评价活动图

时序图设计

图 3 描述了用户进行信用评价的执行序列过程。

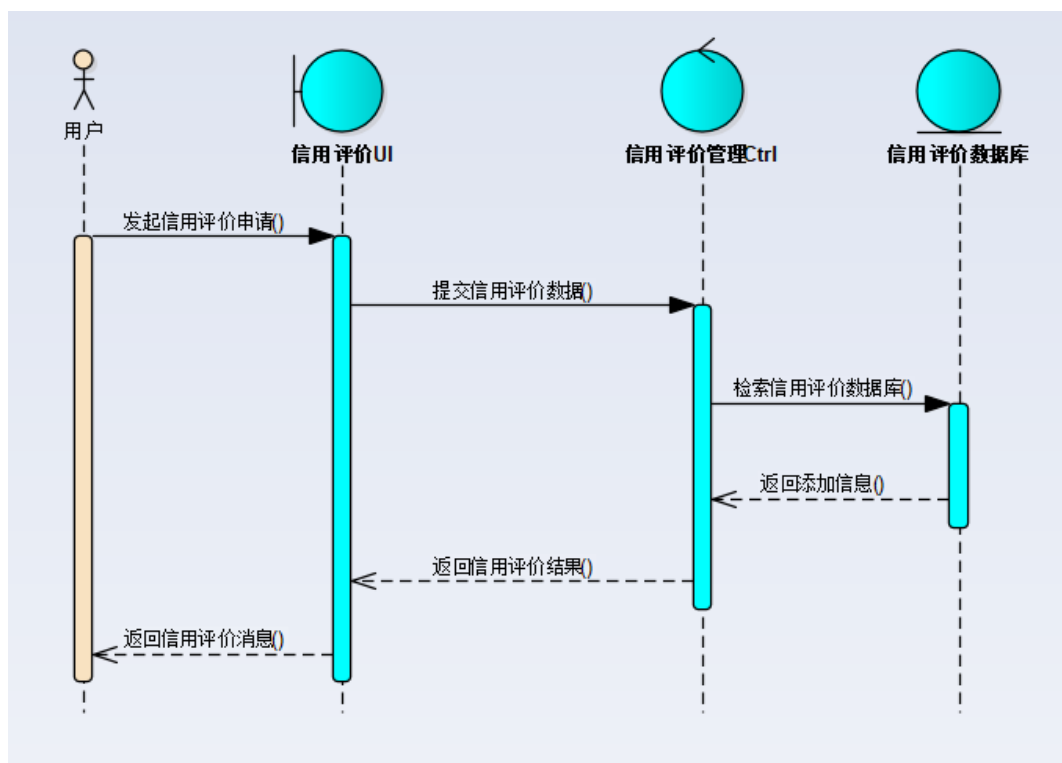


图 3 注册序列图

5. 功能模块设计

具体的功能模块如下：

1. 主界面 home.wxml

调用背景：

- (1) 进入微信小程序时，默认为该页面
- (2) 从其他页面点击返回回到此界面
- (3) 出现未知错误，默认跳转到该页面

页面组成：

主界面主要由上、下两部分组成。上部包含三个按钮，分别为个人中心、搜索栏、新建项目。下部分为参与项目列表，每一部分包含了项目类型以及相对应图片、项目名称、项目截止时间、个人完成情况、项目参加人数。上部固定不动，下部可进行垂直拖拉方便浏览。

调用描述：

(1) 页面调用时，先通过微信小程序请求获取到用户 openid，随后利用 openid 从数据库内请求用户所参与项目，并以列表形式展现在页面中。随后从数据库内获取用户相关消息，若存在新消息，则弹出窗口通知，用户若进入，则跳转至 info_center.wxml 页面。

(2) 用户点击个人中心时，判断用户的登录状态，若用户已注册，则跳转至 personal_center.wxml 页面，若未注册，则跳转到 register.wxml 界面。

(3) 用户点击搜索栏时，跳转至 find.wxml 界面

(4) 用户点击新建项目时，判断用户的登录状态，若用户已注册，则跳转至 new_project.wxml 页面，若未注册，则跳转到 register.wxml 界面。

(5) 用户点击项目列表内项目时，根据所点击项目不同，跳转至不同参数的 project.wxml 页面；若点击项目为已完成项目，则跳转至相应 project_settle.wxml 页面；若点击项目为已结束项目，则跳转至相应的 project_his.wxml 页面。

2. 个人中心 personal_center.wxml

调用背景：

- (1) 在主界面中点击个人中心按钮进入

页面组成：

页面由上下两部分组成，上部为四段文本，展示用户 ID、认证信息、个人简介以及信用等级。下部由一个按钮组成，为修改信息。

调用描述：

(1) 页面调用时，从数据库请求用户的 ID、认证信息、个人简介、信用等级，并展示在页面中。

(2) 用户点击修改信息时，跳转至 info_change.wxml 界面。

3. 修改个人信息页面 info_change.wxml

调用背景：

- (1) 在个人中心中点击修改信息按钮进入

页面组成：

页面包含了一个文本框以及一个提交按钮，文本框中为用户个人信息，用户可进行修改，按下提交按钮进行表单提交。

调用描述：

(1) 用户在点击提交按钮时，对文本内容进行判断，若文本无变化，则不进行返回。若变化，则筛查敏感词。若存在敏感词，则进行 alert 提示，若不存在，则进行表单提交，并返回至 personal_center.wxml

4. 新建项目页面 new_project.wxml

参数：ParentID=? 代表项目的父项目 ID，若为空，则为建立根项目

调用背景：

- (1) 在主界面中点击新建项目按钮进入
- (2) 在项目页面中点击新建子项目进入

页面组成：

页面包含一条展示文本，显示本项目所依赖的父项目或提示本项目为根项目；四个文本框，包括项目名称、项目信息、项目需求、联系方式；一个选择框，选择项目的结束时间，以及一个提交按钮。

调用描述：

(1) 在用户点击提交时，对选择框以及文本框内容进行判断，若存在空内容，则进行 alert 提示相应内容。若不存在非空内容，则将表单提交至服务端，挂起监听，进行转圈提示。

(2) 若得到服务端的 true 反馈，则进行提示申请成功，跳转到相应的 project.wxml 页面。若得到 false 反馈，则提示申请失败。若无法在 10000ms 内得到反馈，则提示请求超时，不进行继续操作。

5. 项目页面 project.wxml

参数：ProjectID=? 代表页面的项目 ID

调用背景：

- (1) 在主界面中点击项目进入
- (2) 新建项目后进入
- (3) 在发现界面中进入

页面组成：

页面包含三个部分：左、中、右。用户可通过滑动的方式在三个部分中进行切换，进入页面时默认为中间页面，包括五条展示文本，显示本项目的父项目、项目名称、项目信息、项目需求、联系方式。并在下方展示参与者头像。

右侧部分为父项目、本项目、子项目以按钮形式列表展示。

左侧部分提供四个按钮，分别为邀请成员、提交项目、新建子项目、退出项目。

调用描述：

(1) 进入页面时，对数据库进行请求，获取用户在项目中的权限，若用户权限为负责人，则将提交项目按钮改名为完成项目、将退出项目改为关闭项目。

(2) 用户点击邀请用户时，小程序生成相应的小程序二维码，即为项目页面邀请链接，展示在界面上。用户再次点击时隐藏。

(3) 用户点击提交项目时，进行 alert 提醒，若用户继续提交，向数据库请求用户在项目中的权限，若用户为普通成员，为提交普通提交申请；若用户为负责人，则提交项目完成申请，向服务器申请提交，挂起监听。若服务器返回 true，则提示提交成功；若返回 false，则提示提交失败；若在 10000ms 内无反馈，则提示提交超时。

(4) 用户点击新建子项目时，进行 alert 提醒，若用户继续新建，则带本项目参数 ProjectID 跳转至 new_project.wxml。

(5) 用户点击退出项目时，进行 alert 提醒，若用户继续退出，向数据库请求用户在项目中的权限，若用户为普通成员，为提交普通退出申请；若用户为负责人，则提交项目关闭申请，向服务器申请提交，挂起监听。若服务器返回 true，则提示退出成功，跳转至 home.wxml 页面；若返回 false，则提示退出失败；若在 10000ms 内无反馈，则提示提交超时。

(6) 用户在右侧部分中点击项目列表时，带参数直接跳转至相应 project.wxml 页面。

6. 发现页面 search.wxml

调用背景：

(1) 在主界面中点击搜索按钮进入

页面组成：

主界面主要由上、下两部分组成。上部包含一个选项框，表示项目类型筛选。下部分为参与项目列表，每一部分包含了项目类型以及相对应图片、项目名称、项目截止时间、个人完成情况、项目参加人数。上部固定不动，下部可进行垂直拖拉方便浏览。

调用描述：

(1) 页面调用时，从数据库内请求用户所参与项目，并以列表形式展现在页面中。

(2) 选项框数值变动时，根据相应条件进行筛选，将筛选结果以列表形式展现在页面中。

(3) 用户点击项目列表内项目时，根据所点击项目不同，跳转至不同参数的 project.wxml 页面。

7. 消息中心页面 info_center.wxml

调用背景：

(1) 在主界面中获取消息推送后点击进入

页面组成：

页面为一滚动列表，列表中展示用户相关信息。

调用描述：

(1) 页面调用时，从数据库获取用户相关消息列表，并展示在页面中。

(2) 用户点击消息时，跳转到相应 project.wxml 页面。

8. 项目结算页面 project_settle.wxml

参数：ProjectID=? 代表页面的项目 ID

调用背景：

(1) 在主界面中，负责人点击已完成项目进入

页面组成：

页面为一列表，展示所有项目内参与成员，负责人可对每名成员进行打分，最终算出加权比例。下部为一提交按钮。

调用描述：

(1) 页面调用时，从数据库获取项目成员列表，并展示在页面中。

(2) 用户点击提交时，进行提示，若确认提交，则向服务器申请提交，挂起监听。若服务器返回 true，则提示提交成功；若返回 false，则提示提交失败；若在 10000ms 内无反馈，则提示提交超时。

6. 存储过程设计

小程序总共涉及 30 余个存储过程，在此以两个主要存储过程为例：

1. getSuperiors()

功能描述：查询所有父节点

入口参数：@uid (int, 无默认值)

出口参数：@superiors (varchar(1000))

程序实现：

```
CREATE DEFINER=`root`@`localhost` FUNCTION `getSuperiors`(`uid` int) RETURNS
varchar(1000) CHARSET gb2312
BEGIN
    DECLARE superiors VARCHAR(1000) DEFAULT '';
    DECLARE sTemp INTEGER DEFAULT uid;
    DECLARE tmpName VARCHAR(20);

    WHILE (sTemp>0) DO
        SELECT parentId into sTemp FROM node where uid = sTemp;
        SELECT name into tmpName FROM node where uid = sTemp;
        IF(sTemp>0)THEN
            SET superiors = concat(tmpName,',',superiors);
        END IF;
    END WHILE;
    SET superiors = LEFT(superiors,CHARACTER_LENGTH(superiors)-1);
    RETURN superiors;
END
```

2. getSubordinate()

功能描述：查询所有子节点

入口参数：@uid (int, 无默认值)

出口参数：@superiors (varchar(1000))

程序实现：

```
CREATE DEFINER=`root`@`localhost` FUNCTION `getSubordinate`(`uid` int) RETURNS
varchar(2000) CHARSET gb2312
BEGIN
    DECLARE str varchar(1000);
    DECLARE cid varchar(100);
    DECLARE result VARCHAR(1000);
    DECLARE tmpName VARCHAR(100);
    SET str = '$';
    SET cid = CAST(uid as char(10));
    WHILE cid is not null DO
        SET str = concat(str, ',', cid);
```

```

SELECT group_concat(userId) INTO cid FROM node where FIND_IN_SET(parentId,cid);
END WHILE;
SELECT GROUP_CONCAT(name) INTO result FROM node WHERE
FIND_IN_SET(parentId,str);
RETURN result;
END

```

7. 接口设计

7.1 用户接口

以微信小程序为平台，使用类 HTML、CSS、JS 进行前端构建，为用户提供简洁、便利的操作体验。

7.2 外部接口

数据储存：本小程序使用 PHP 与 MySQL 数据库作为后端进行数据交互。

7.3 内部接口

小程序为数据为中心，服务端利用 PHP 与数据库进行交互

8. 角色授权设计

角色授权见下表：

子系统名称	管理员	项目发起人	项目参与者
用户信息管理系统	●	●	●
项目发起系统		●	
项目追踪管理系统	○	●	○
项目结算系统		●	○
历史查看系统	●	●	●
信用评价系统		●	○
消息提醒系统		●	●
发现系统	○		●

9. 系统错误处理

9.1 出错信息

- (1) 对各项需用户填写信息，进行严格的有效性、安全性检查，减少错误发生几率。
- (2) 对程序运行中的异常进行捕获，并用统一的方式将错误信息反馈给用户，并提交给服务端便于后期维护
- (3) 制作统一的错误反馈 alert，使用户在进行错误访问后能够获得提示。

9.2 故障预防与补救

以统一的机制进行网站权限的控制，对程序中用到的数据尽量进行加密，以减少黑客攻击的危害。定期对数据库中的数据进行海量备份及增量备份。

9.3 系统维护设计

(1) 编码实现时应采用模块化和分层的思想，提高模块内部的内聚，减少模块间的耦合。使系统逻辑结构清晰，从而增强可读性和可维护性。

(2) 面向数据与面向对象相结合，模块划分符合面向对象思想。

(3) 在编码过程中注意标识符命名的意义，添加适量注释。

10. 测试计划

测试计划作为本系统的测试知道，测试成员按计划 and 规定进行测试。本系统的测试将从单元测试、集成测试、系统测试 3 个方面来设计。