

# 系统设计和数据库设计说明

打代码一定要笑

# 目录

PART ONE

系统设计说明书

PART TWO

数据库设计说明书

PART THREE

问题解答

PART FOUR

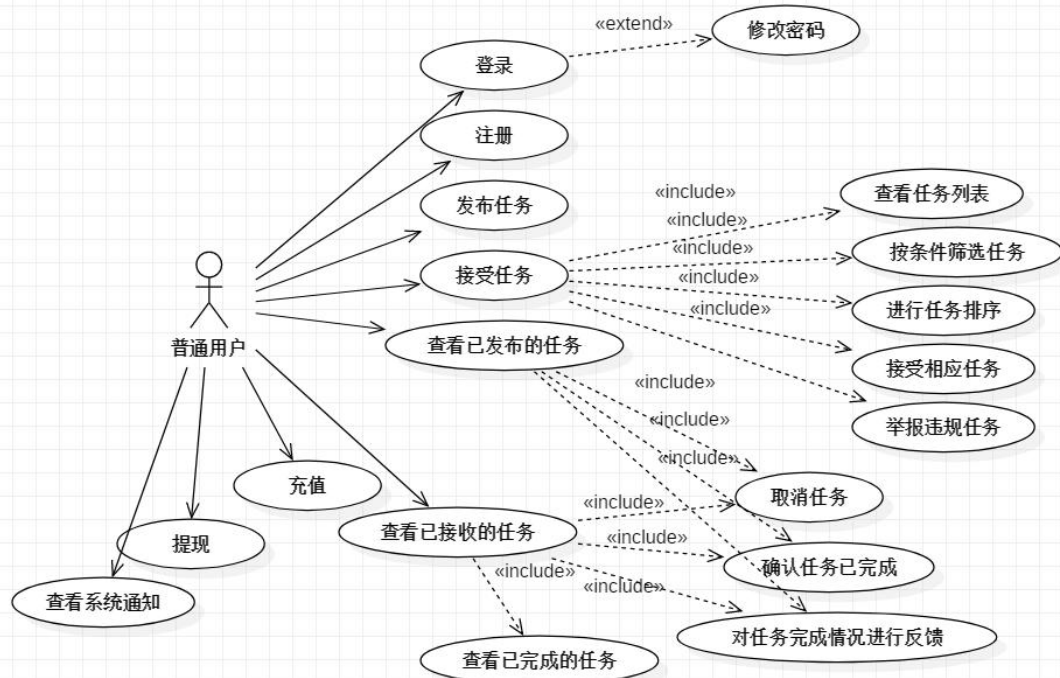
小组分工



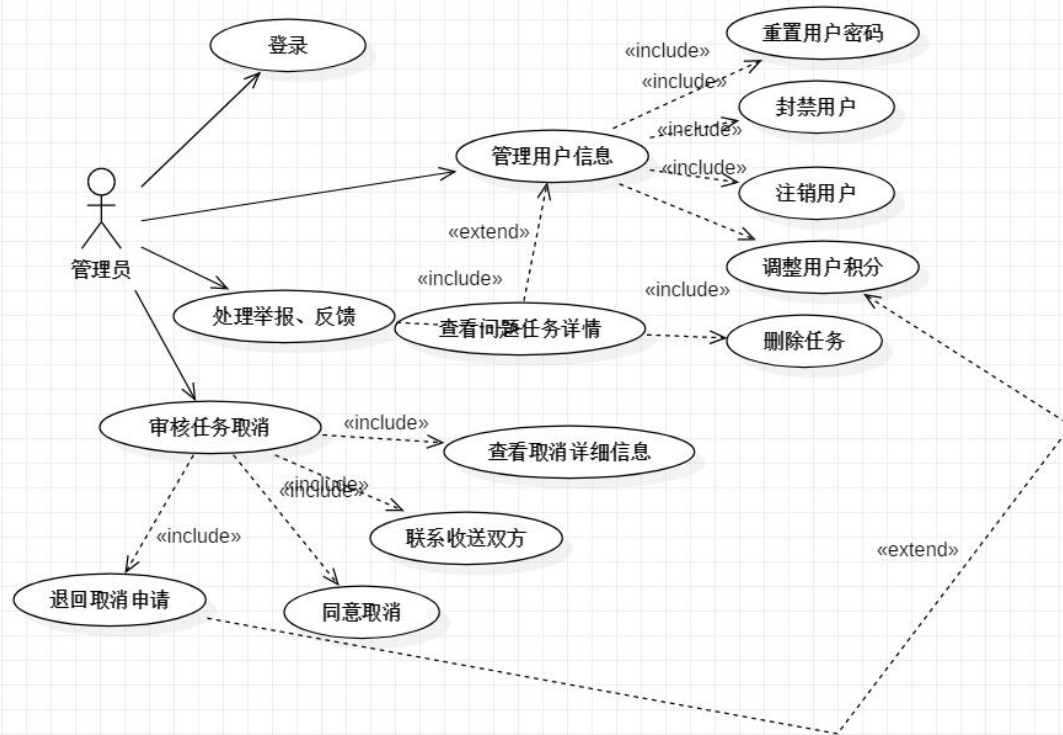
# PART One

▼  
系统设计说明

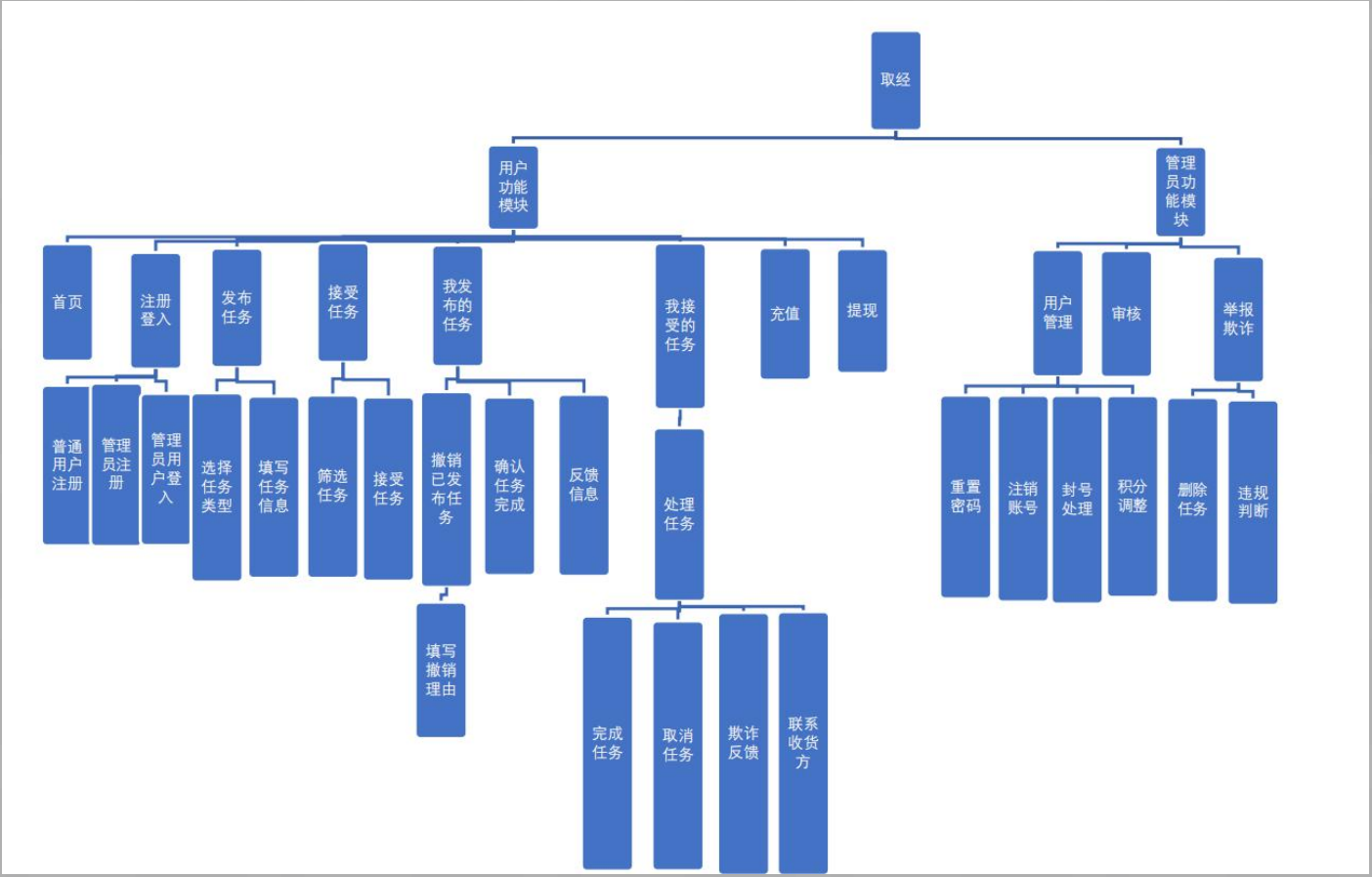
# 体系结构设计



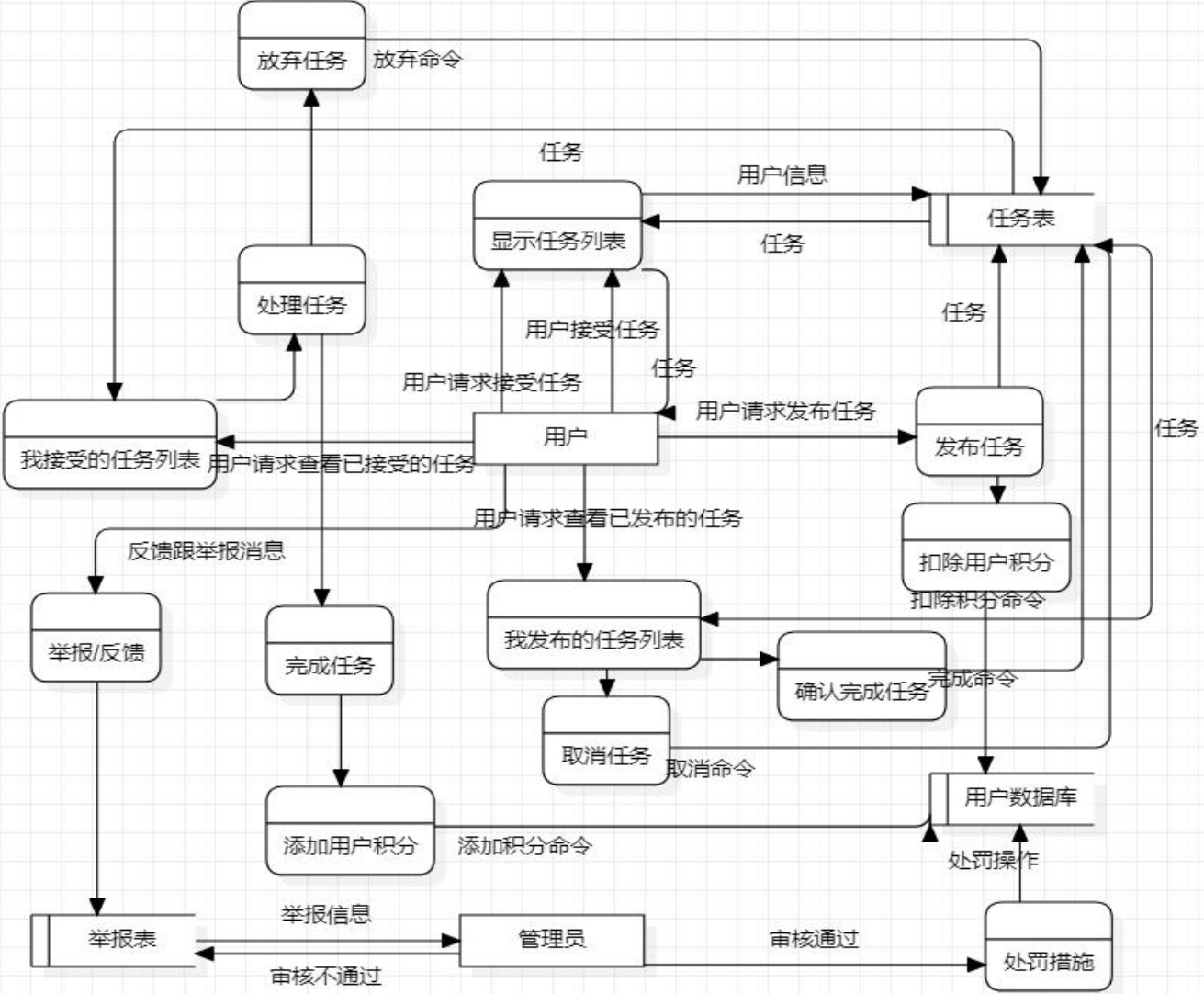
# 体系结构设计



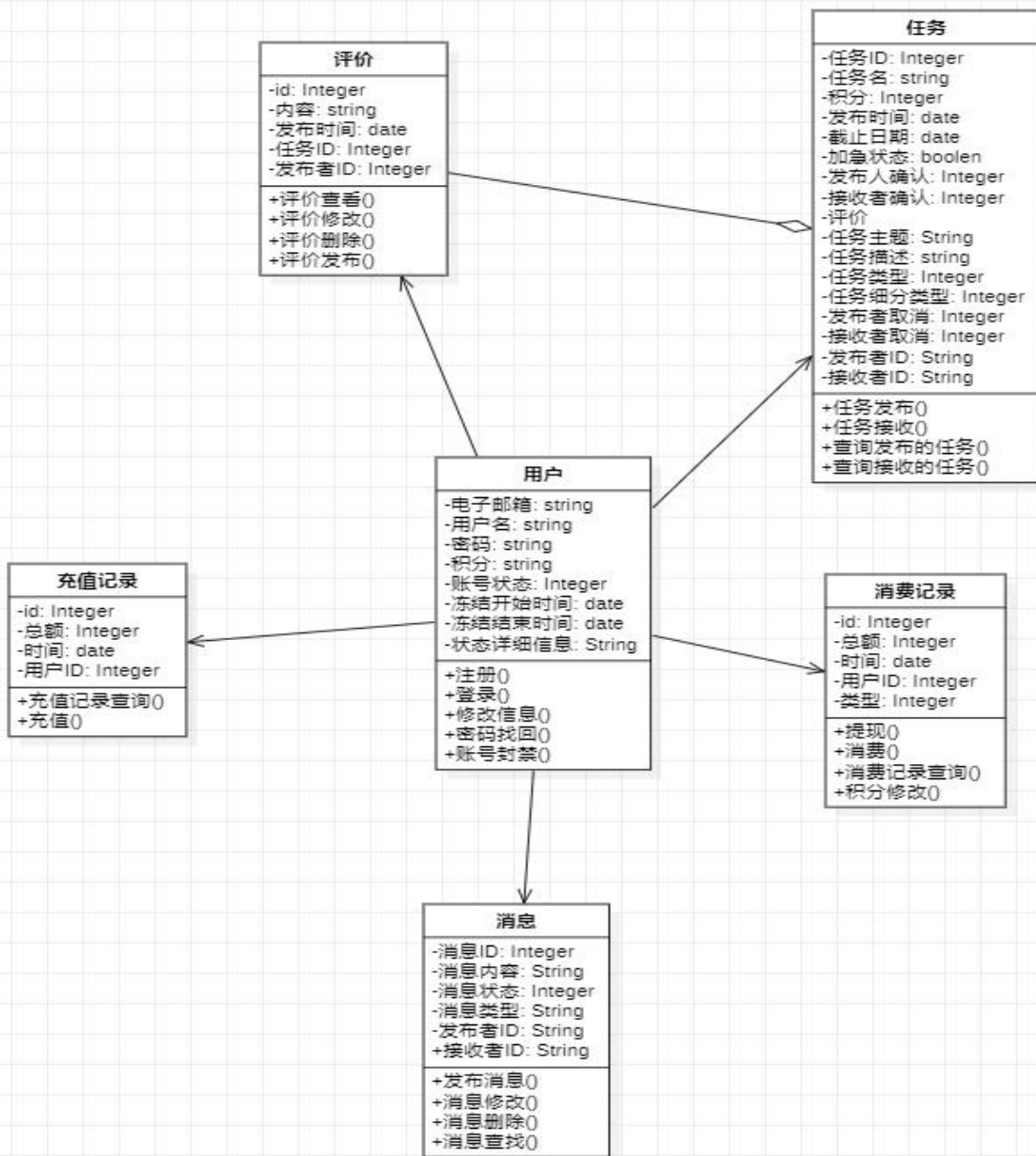
功能模块层次图



核心数据流图

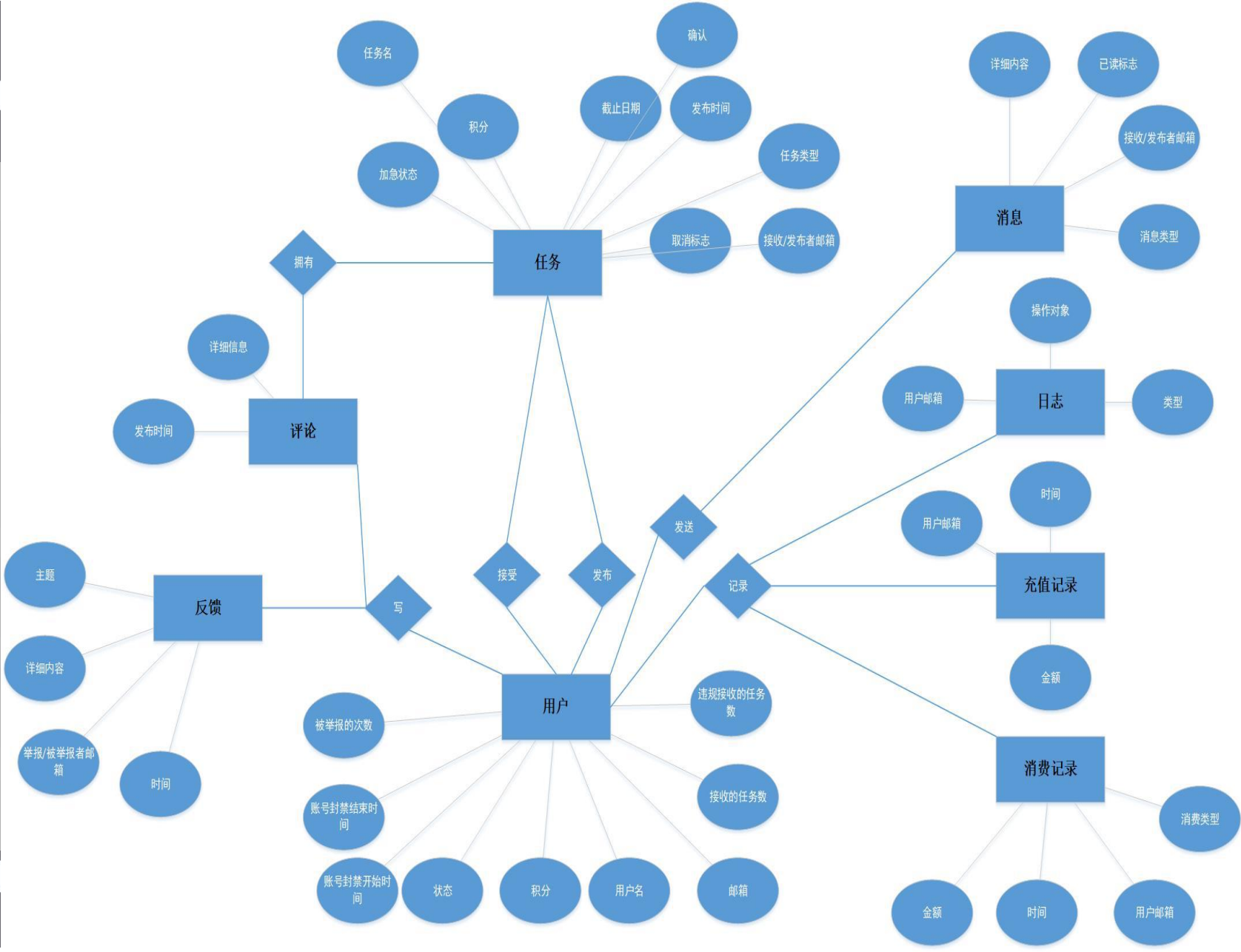


# 用户类图





# ER分析



# 用户功能模块接口设计

URI	接口功能	请求方法	请求体
a/register	用户注册	Post	Name Email Password
a/validEmail	验证用户邮箱	Post	Email
a/getUserInfo	获得用户数据	Get	
a/login	用户登入	Post	Email

			Password
a/code	修改用户密码	Put	Password
a/u/message	修改用户信息	Put	User_name User_Content
a/u/listTask	获取任务列表	Get	Page Type NextType
a/u/releaseTask	发布任务	Post	Task_name Task_detail Token Deadline Task_type Task_point

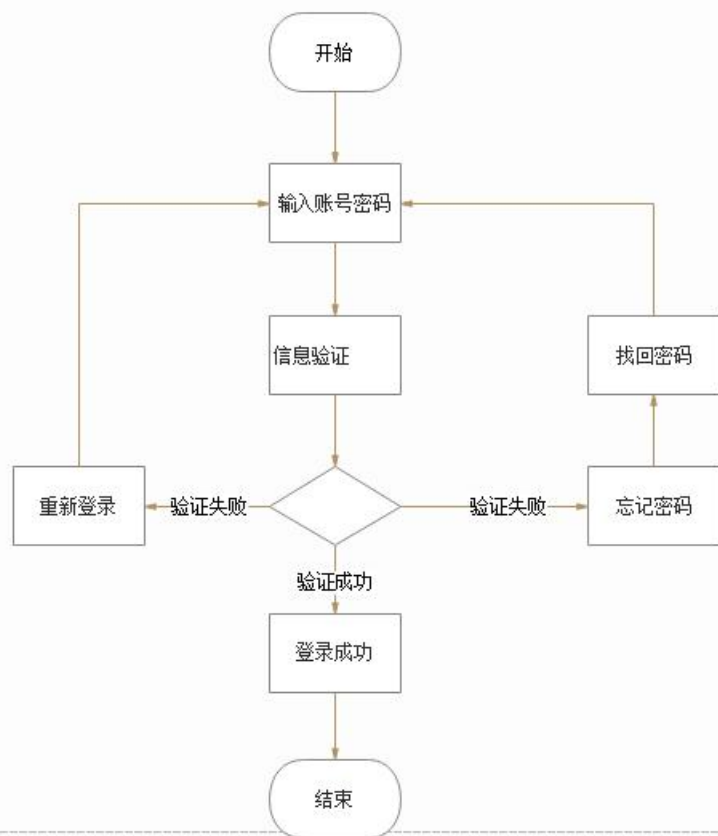
a/u/acceptTask/{id}	接受任务	Post	Task_Email
a/u/listTask/{email}	按 特 定 email 筛选任务	Get	Page State
a/u/deleteTask/{id}	撤销指定 id 的任务	Put	
a/u/commentTask/{id}	对指定 id 的任务进行评论	Post	Theme Content
a/u/finishTask/{id}/	完成指定 id 的任务	Put	
a/u/cancelTask/{id}	放弃指定 id 的任务	Put	
a/u/feedBack	反馈问题	Post	Theme Content ReportedEmail
a/u/sendMessage	发送消息	Post	Content MessageType
a/u/listMessage/{email}	获 取 指 定 email 的 消息	Get	Page
a/u/recharge	充值	Post	Amount
a/u/consume	消费	Post	Amount Type
a/u/listRecharge	查询充值记录	Get	Email Time
a/u/listConsumption	(根据时间/消费类型)查询消费记录	Get	Email Time Type
a/u/logout	注销	Get	

管理员功能模块接口设计

URI	接口功能	请求方法	请求体
b/register	管理员注册	Post	Username Password Code
b/login	管理员登录	Post	Username Password
b/u/ <u>updateUser</u> / <u>{email}</u>	更新用户信息	Put	<u>UserName</u> Points State <u>StartTime</u> <u>EndTime</u>
b/u/ <u>updateTask</u> / <u>{id}</u>	更新任务信息	Put	<u>TaskName</u> <u>TaskState</u> .....
b/u/ <u>listUser</u>	查看用户信息	Get	Email <u>UserName</u> State
b/u/ <u>listTask</u>	查看用户任务信息	Get	<u>TaskName</u> Type <u>NextType</u> <u>publisherEmail</u> <u>recipientEmail</u>
b/u/ <u>listRecharge</u> / <u>{email}</u>	查询根据指定email充值记录	Get	
b/u/ <u>listConsumption</u> / <u>{email}</u>	查询根据指定email消费记录	Get	
b/u/ <u>sendMessage</u> / <u>{email}</u>	向指定的用户发送消息	Post	Theme Content
b/u/ <u>listFeedback</u>	查看问题反馈	Get	Page
b/logout	注销	Get	

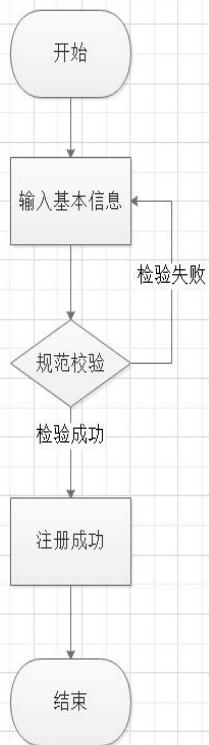
# 活动图

登录



# 活动图

注册

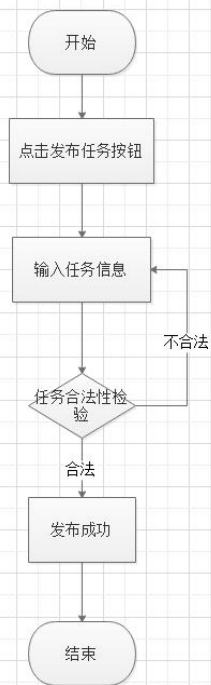


找回密码

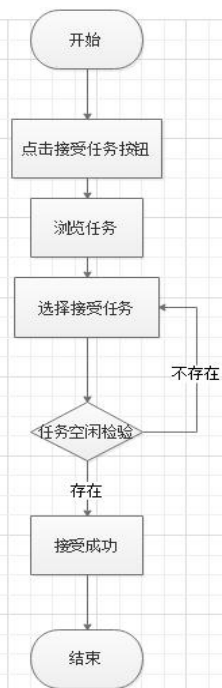


# 活动图

## 发布任务

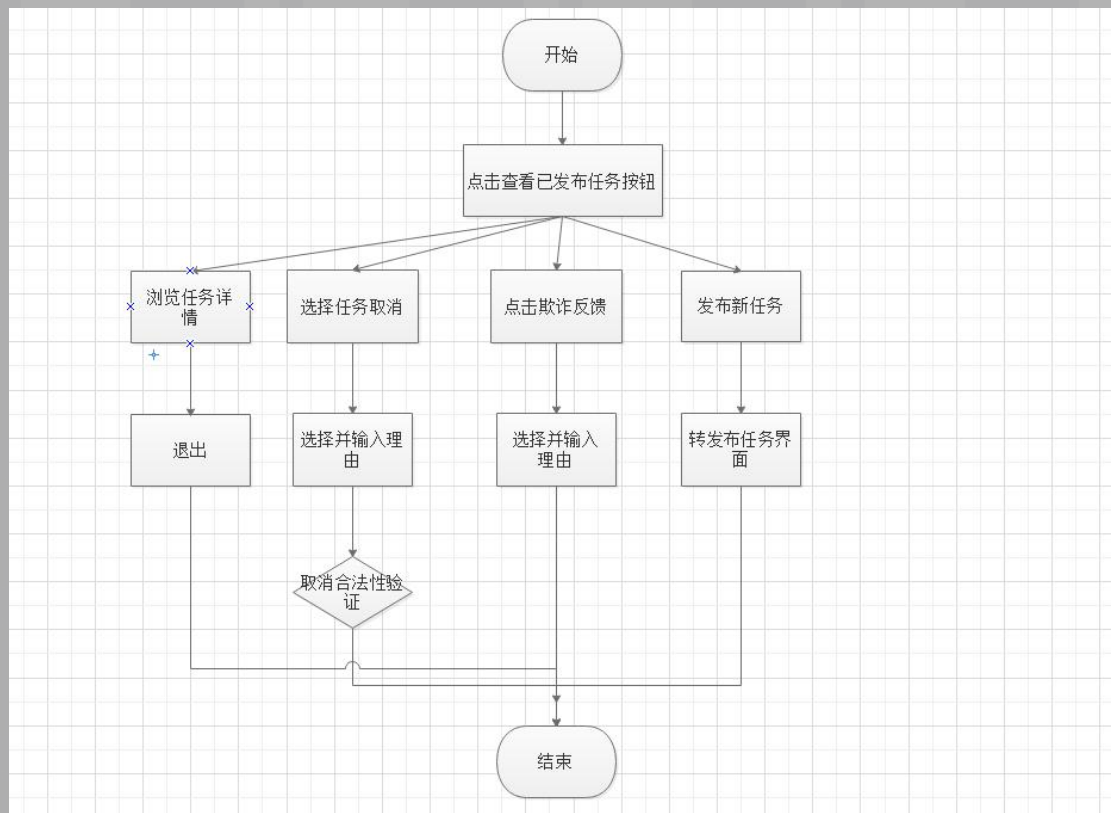


## 接受任务



# 活动图

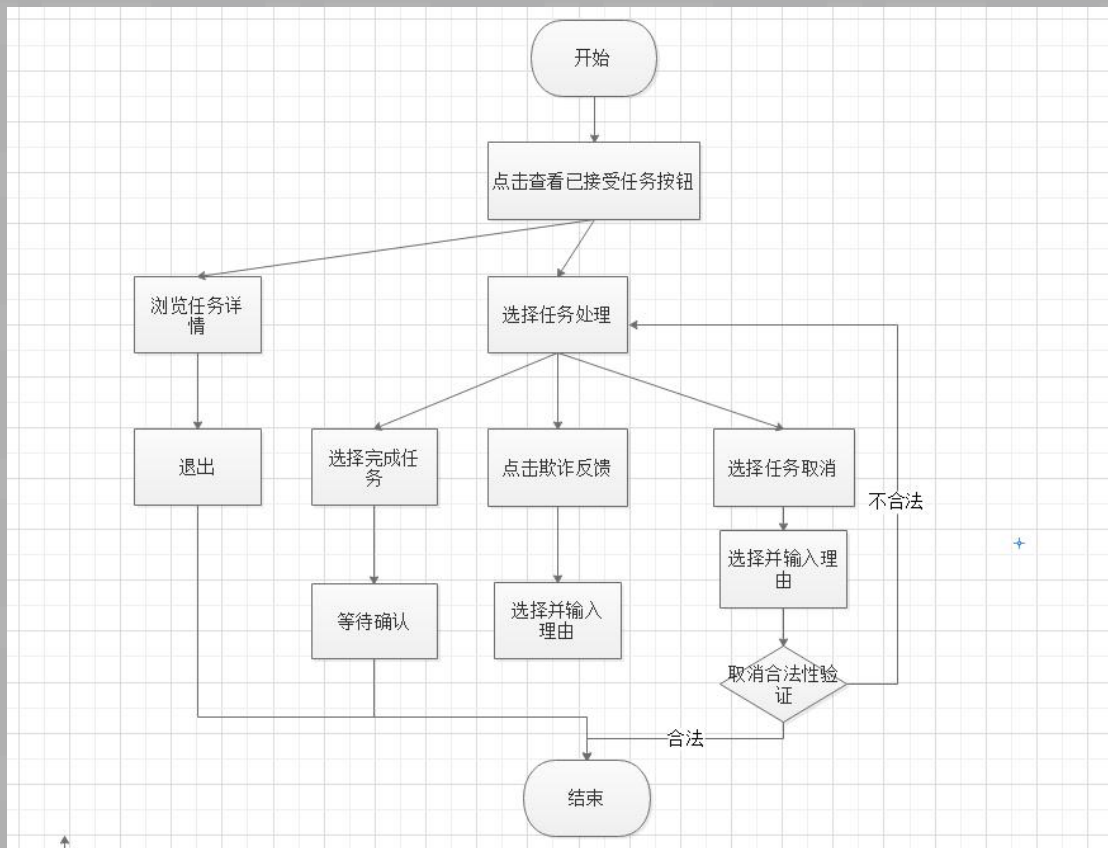
查看已发布任务





# 活动图

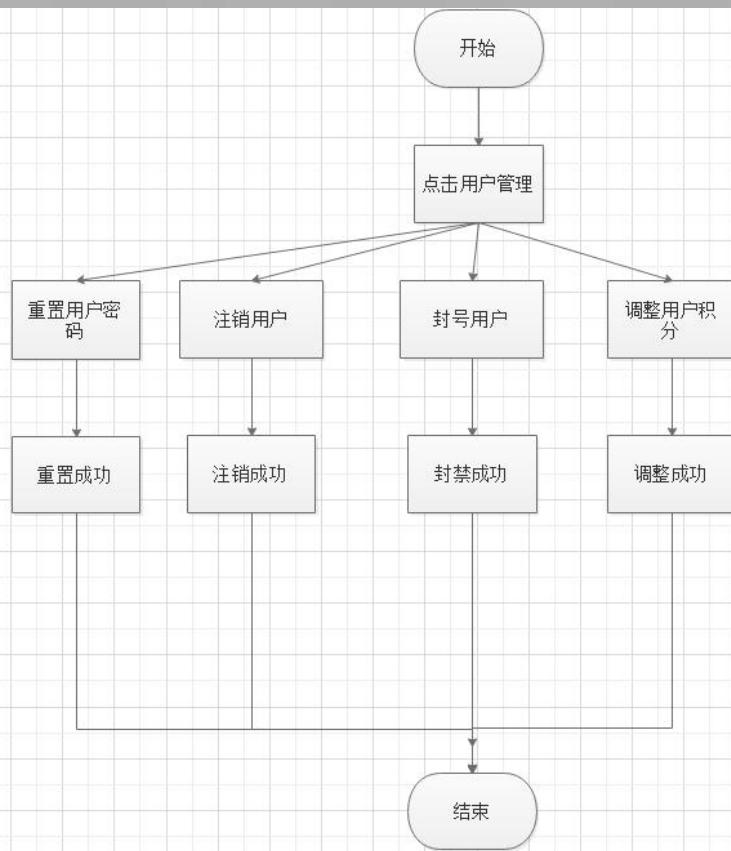
查看已接受任务





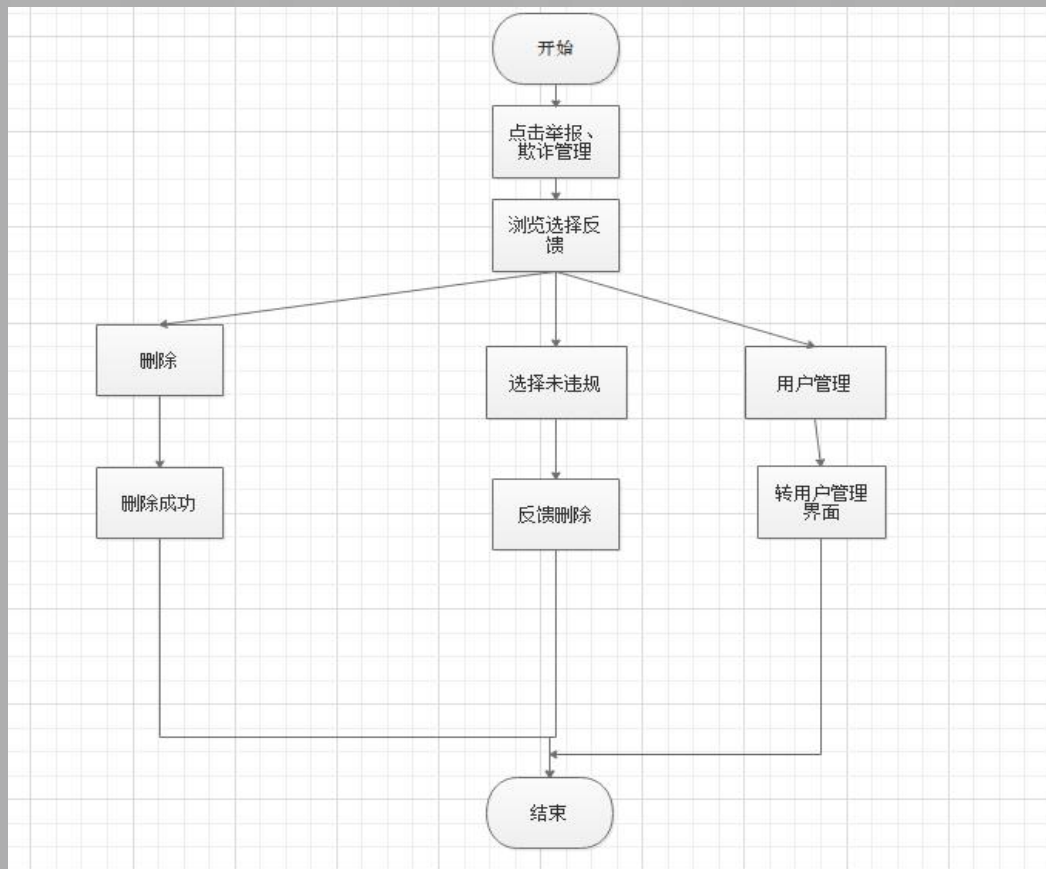
# 活动图

## 用户管理



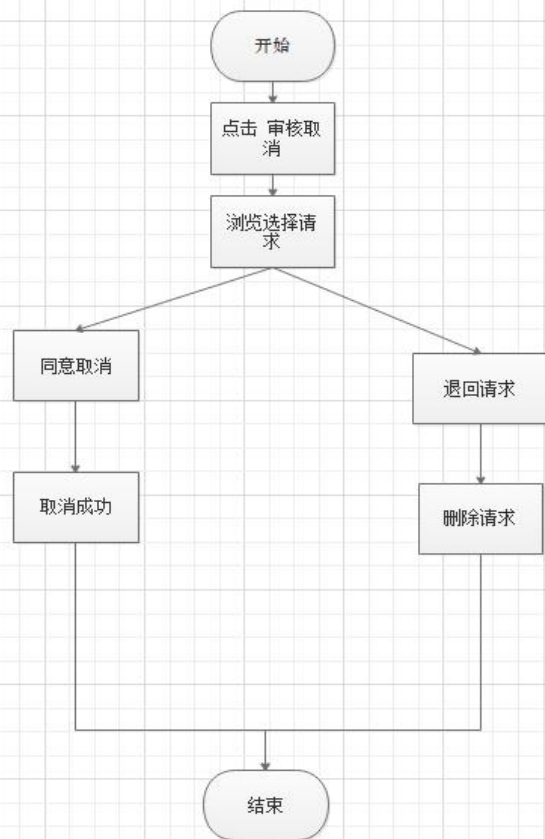
# 活动图

## 欺诈举报处理



# 活动图

审核取消任务





# PART Two



---

数据设计说明





数据库设计说明书  
数据库类型:MySQL  
数据库名称:qujin

在本系统中，数据库的设计采用Visio进行，并且采用面向对象的设计方法，首先进行对象实体的设计，最后将对象持久化到数据库中,所有的表和表之间的关联(ER图)都采用标准的Visio设计工具进行,这样能够将整个系统的设计和数据库设计有机的结合起来。

# Administrator表：存储管理员账号的信息

名	类型	长度	小数点	不是 null	
username	varchar	20	0	<input checked="" type="checkbox"/>	1
password	varchar	20	0	<input type="checkbox"/>	

## Userstate表：

id	name
1	冻结
2	激活

# User表：存储用户账户的信息

Name	Type	Length	Decimals	Allow Null	
email	varchar	20	0	<input type="checkbox"/>	1
IDNumber	varchar	11	0	<input type="checkbox"/>	2
username	varchar	20	0	<input checked="" type="checkbox"/>	
password	varchar	20	0	<input checked="" type="checkbox"/>	
points	int	11	0	<input checked="" type="checkbox"/>	
state	int	11	0	<input checked="" type="checkbox"/>	
content	text	0	0	<input checked="" type="checkbox"/>	
startTime	datetime	0	0	<input checked="" type="checkbox"/>	
endTime	datetime	0	0	<input checked="" type="checkbox"/>	
receiveTaskNumber	int	11	0	<input checked="" type="checkbox"/>	
maliciousAcceptanceNumber	int	11	0	<input checked="" type="checkbox"/>	
reportedNumber	int	11	0	<input checked="" type="checkbox"/>	

points:用户所拥有的积分,默认值:0  
state:用户账户的状态,默认值:1,外键  
content:用户状态的详细信息  
startTime:用户处于某种状态的开始时间  
endTime:用户处于某种状态的结束时间  
receiveTaskNumebr: 用户已接收任务数,用于判定是否可以继续接受任务  
maliciousAcceptanceNumber: 违规接受任务数,保存用户未按规定完成任务的数量  
reportedNumber: 该用户被成功举报的数量

# task表： 存储任务信息

Name	Type	Length	Decimals	Allow Null	
id	int	11	0	<input type="checkbox"/>	1
name	varchar	20	0	<input type="checkbox"/>	
points	int	11	0	<input checked="" type="checkbox"/>	
deadline	datetime	0	0	<input type="checkbox"/>	
confirmDeadline	datetime	0	0	<input checked="" type="checkbox"/>	
expedited	int	1	0	<input type="checkbox"/>	
senderAccomplish	int	1	0	<input type="checkbox"/>	
receiverAccomplish	int	1	0	<input type="checkbox"/>	
commentId	int	11	0	<input checked="" type="checkbox"/>	
content	text	0	0	<input checked="" type="checkbox"/>	
taskType	int	11	0	<input type="checkbox"/>	
taskNextType	int	11	0	<input checked="" type="checkbox"/>	
senderCancle	int	1	0	<input type="checkbox"/>	
receiverCancle	int	1	0	<input type="checkbox"/>	

## comment表：

名	类型	长度	小数点	不是 null	
id	int	11	0	<input checked="" type="checkbox"/>	1
content	text	0	0	<input type="checkbox"/>	
time	datetime	0	0	<input type="checkbox"/>	

publishTime:发布时间  
comfiredDeadline: 雇主确认收货的时间  
deadline:任务结束时间  
expedited:是否加急,默认值:0.(0:为加急,1:加急)  
publisherAccomplish:发布者确认,默认值:0(0:未确认,1:确认)  
recipientAccomplish:接收者确认,默认值:0(0:未确认,1:确认)  
commentId:任务反馈信息,外键  
theme:任务主题  
content:任务详细内容  
taskType:任务类型,外键  
taskNextType:任务类型细分  
publisherCancle:发布者取消任务, 默认值:0(0:未取消,1:取消)  
recipientCancle:接收者取消任务, 默认值:0(0:未取消,1:取消)

## expedited表：

id	name
1	未加急
2	加急

## taskType表：

id	name
1	快递代取
2	文件代送
3	文件代取
4	食堂代买
5	物品代购
6	其他

# message表:存储系统、用户之间的消息

名	类型	长度	小数点	不是 null	
id	int	11	0	<input checked="" type="checkbox"/>	1
content	text	0	0	<input type="checkbox"/>	
messageState	int	11	0	<input checked="" type="checkbox"/>	
messageType	int	11	0	<input checked="" type="checkbox"/>	
recipientEmail	varchar	20	0	<input checked="" type="checkbox"/>	
publisherEmail	varchar	20	0	<input checked="" type="checkbox"/>	

messageState: 是否已读, 默认值: 0, (0: 未读, 1: 已读)

messageType: 消息类型, 外键  
recipientEmail: 接收消息的用户邮箱, 外键

publisherEmail: 发送消息的用户邮箱 (系统消息为 admin).

## messageType表:

id	name
1	系统通知
2	用户消息



feedback表:存储举报、问题反馈信息

栏位	索引	外键	触发器	选项	注释	SQL 预览
名			类型	长度	小数点	不是 null
id			int	11	0	<input checked="" type="checkbox"/> 1
theme			varchar	255	0	<input checked="" type="checkbox"/>
content			text	0	0	<input checked="" type="checkbox"/>
reportEmail			varchar	20	0	<input checked="" type="checkbox"/>
reportedEmail			varchar	20	0	<input checked="" type="checkbox"/>
time			datetime	0	0	<input checked="" type="checkbox"/>


expensesRecord表:消费记录

名	类型	长度	小数点	不是 null	
id	int	11	0	<input checked="" type="checkbox"/>	1
amount	int	11	0	<input checked="" type="checkbox"/>	
time	date	0	0	<input checked="" type="checkbox"/>	
userEmail	varchar	20	0	<input checked="" type="checkbox"/>	
type	int	11	0	<input checked="" type="checkbox"/>	


consumptionType表:存储消费类型

id	name
1	发布任务
2	提现
3	惩罚

recharge表:存储充值记录

名	类型	长度	小数点	不是 null	
id	int	11	0	<input checked="" type="checkbox"/>	 1
amount	int	11	0	<input checked="" type="checkbox"/>	
time	datetime	0	0	<input checked="" type="checkbox"/>	
userEmail	varchar	20	0	<input checked="" type="checkbox"/>	

log表： 存储系统日志

Name	Type	Length	Decimals	Allow Null	
id	int			<input checked="" type="checkbox"/>	 1
userEmail	varchar	20	0	<input checked="" type="checkbox"/>	
action	int	11	0	<input checked="" type="checkbox"/>	
taskId	int	11	0	<input checked="" type="checkbox"/>	

Logtype表： 保存操作类型

id	type
1	发布
2	收货方确认
3	送货方确认
4	收货方取消
5	送货方取消
6	接受



# PART Three



---

问题解答



1

考虑增加。可以分出代拿快递任务，食堂代拿外卖任务，代买物品任务，代送文件任务。快递任务里再细分出顺丰快递，韵达快递等，食堂可以细分为哪个食堂，代买任务可以细分出三区代买，四区代买。

2

先发布后审核，设计敏感词拦截大部分违规的任务，还会添加举报功能，对敏感词未拦截到的任务进行举报，举报成功可以获得奖励积分（从发布者那里扣除），成功与否由后台审核，每人每天有5次举报机会，错误举报次数大于5次，将不再可以举报，同样的，被举报人如果被举报成功3次，管理员就会酌情封号。（考虑如果有人闲得无聊，发布的任务，简介是违规的，详细内容是正常的，即钓鱼标题党，也直接会举报成功）

### 3

有，如果接受了任务，截止时间内（考虑到是雇员拿快递途中，接受任务，进行帮拿）没有点击完成提交，系统会推送消息提醒，再两个小时（考虑接受了任务就忘记了，没拿快递，还要去快递站拿）没完成，就系统判定为恶意接受，扣除一定积分。**3**次以上恶意接受，并收到他人的反馈或举报，管理员进行封号审核并处理（一旦被封号处理，不会接受复查）。如果对系统的恶意接受判定有异议，需要提供理由进行管理员复查。（不提供照片或者其他实质性证据只进行文字描述，就不会进行管理员复查）同时，每个人最多只能接受**2**个任务（也是为了防止恶意接受）。

## 4

雇主取消未接受的任务（不能取消已接受的任务，保障雇员的利益），有以下几个常用理由

- 1.我自己完成了。
- 2.太久没人接受。

雇员取消接受了但未完成的任务（出现5次以上取消已接收任务的行为并被反馈问题，将进行管理员审核封号）

- 1.有突发情况，无法完成（扣除积分）
- 2.我忘记了，但是现在不想去完成了（扣除积分）
- 3.有情况发生，却无法联系到雇主（需要提供聊天截图，否则无法提交成功，管理员查看截图，如果发现图文不符等错误，直接封号（意图欺诈管理员，直接封号））

原则上，我们希望的是接受的任务都是能完成的，如果不能完成就不应该接受，接受了无法完成，无论出于何种原因，都应该收到惩罚，与第3点不同的是，第3点是有人无法完成任务却没有及时取消任务导致雇主的利益收到更长时间的侵害，需要人工复查的条件就更苛刻

5

雇员提交了完成，雇主那边系统会推送消息，进行提醒。跟雇员一样，雇主也会有两个小时的确认时间（考虑快递拿到雇主宿舍，雇主不在），如果时间到了却没有确认完成，就会进行强制结算，把积分扣除加给雇员，雇主不可申诉。（如果雇员是没有完成却提交，雇主是可以用反馈系统进行反馈，这不是该题的问题，别杠）



## 6

### 雇员反馈：

- 1.发布任务不存在（扣除双倍积分给雇员）
- 2.发布任务不存在，且无法联系到发布者（扣除双倍积分给雇员，同时如果管理员也无法联系上发布者（24小时没回复），暂时封号，直到修改联系方式）
- 3.发布任务与实际严重不符（扣除双倍积分，同时对雇主进行警告处理，3次警告封号）  
以上的如果雇主积分不足，全部扣除，如果有积分入账，优先偿还欠缺的积分。

### 雇主反馈：

- 1.任务没完成去提交了完成任务（需要雇主提交一些截图或者照片证据，如快递还未被拿的快递信息截图，系统会提醒雇员也提交一些截图或者提交任务的图片或者截图，因为实名制且该问题涉及物品丢失会有刑事风险，学生大概率不敢造假，两方各执一词时，会联系辅导员及警方）
- 2.有完成但是任务超时（只支付一半的积分，与第三点的恶意接受相呼应）

7

有意向，希望与福大助手合作。如果做不到，铁华组的i食堂订餐软件也可以与之合作，同学通过他们的软件下单后，食堂的商家作为发布者（食堂任务只有认证后的食堂商家可以发布，且接受任务的截止时间只有**15分钟**，完成任务的时间只有**30分钟**（考虑吃饭很急），只有中午**12:00-1:15**的午餐和晚上**5:00-6:45**的晚餐可以发布（防止商家的误操作））在我们的平台上发布帮取餐任务，同学们顺路完成任务。

发布时点一下加急就可以设定加急。为了防止出现老师说的全部人都把任务设置成加急的情况，加急之后，只有30分钟的接受任务的时间，如果超时，就是自动下架该任务并发布消息提醒发布者。加急之后的积分不做要求（加急的任务也有工作量大小的区别，无法提前量化）。

不采纳，我们增加了新增举报功能，用以找到一些有意或无意识的错误。同时有取消任务作为第二重保障，但是欺诈还是有可能存在的，三者的性质各不相同，不只是只是文字的区别，举报是群众性的对违规内容的举报，取消是还没有付出劳动之前的自己保障，欺诈是雇员付出劳动之后的系统反馈保障机制。我们的用意在于震慑不良用户，规范用户的使用，从而提高产品的寿命。

组合排序，我们目前有加急，截止时间，积分三个排序。比如，先点击加急，就会用加急作为排序的主排序，再点击积分，就会把加急的全部按照积分进行排序。



# PART Four



---

小组分工



## 小组分工

柯燊熙：PPT设计，博客，说明书的检查

洪澍楠：QA的记录

欧振贵：数据库设计说明书

麻继友：数据库设计说明书

沈泳焕：系统设计说明书

林昌锴：系统设计说明书



谢谢大家