

# High-level Design (概要设计规约)

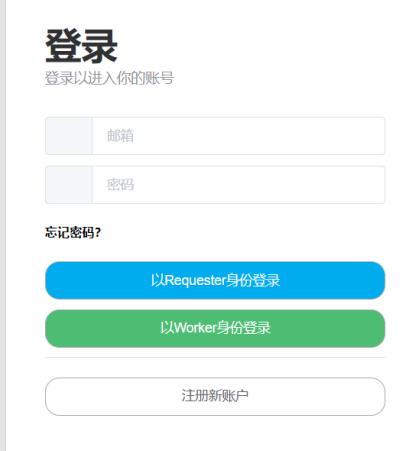
## Prototype Design (原型设计)

The UI design of the project

### Tourist Section

Sign in

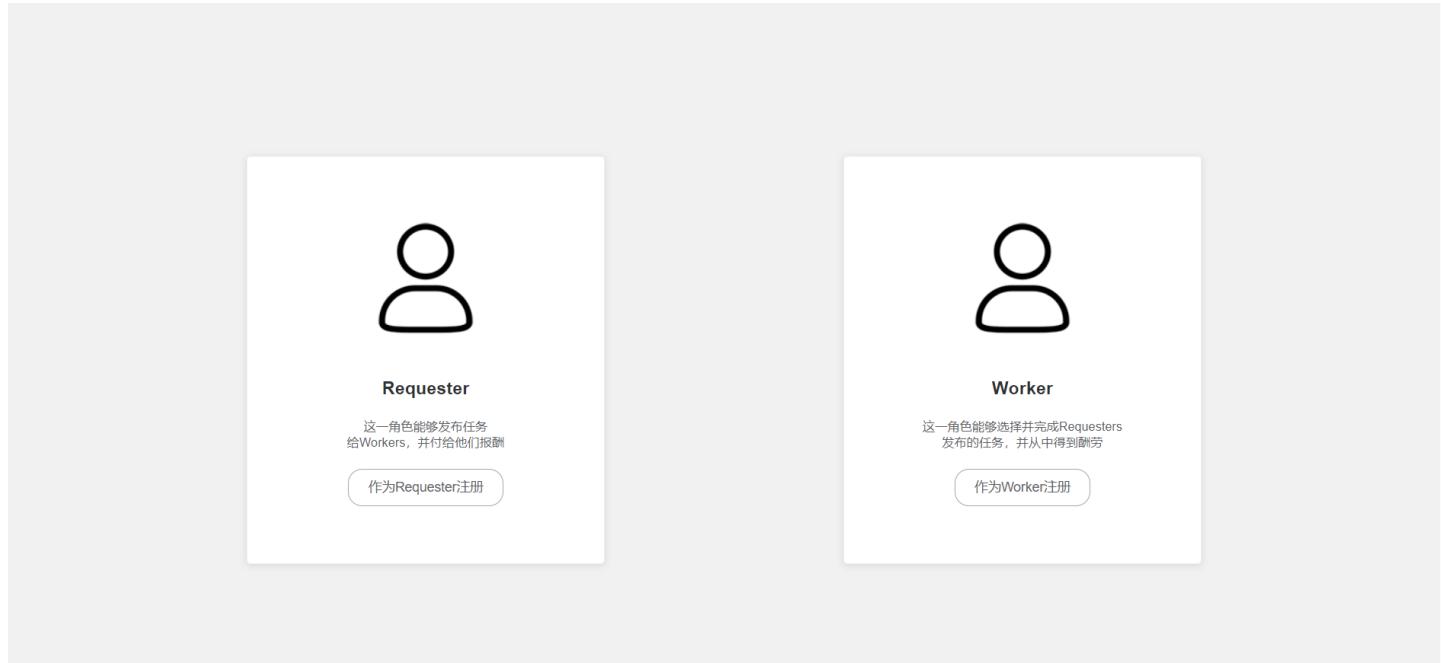
This part is for tourist to sign in.



A wireframe prototype of a login form titled "登录" (Login). It includes fields for "邮箱" (Email) and "密码" (Password), a "忘记密码?" (Forgot Password?) link, and two large buttons: a blue one labeled "以Requester身份登录" (Log in as Requester) and a green one labeled "以Worker身份登录" (Log in as Worker). A "注册新账户" (Register New Account) button is also present.

Sign up

This part is for tourist to sign up, and can choose the identity.



# 注册

创建新账号

用户名

邮箱

密码

确认密码

**创建新账号**

已经拥有账户? [登录](#)

Homepage

Learn about this platform, get platform introductions and pricing rules.

The image shows a top-down view of a wooden desk. On the desk, there is an open laptop with its screen partially visible, a white cup of coffee, and several papers. One paper has a pie chart titled "Customer Demographic Distribution" with segments for "50岁" (50 years old), "30岁" (30 years old), and "15岁" (15 years old). Another paper is a "Trading Graph" with a line chart showing trends over time. A third paper is a news clipping from "Wall Street Journal" dated "Monday Jan 14, 2018". The laptop screen shows some code or data analysis software. In the top left corner of the image area, there is a navigation bar with Chinese text: "主页" (Home), "简介" (Introduction), and "定价" (Pricing). In the top right corner, there is a "登录" (Login) button. Overlaid on the center of the image is the TJ众测 logo, which consists of two stylized eyes and the text "TJ众测". Below the logo is a white button with the text "现在注册加入TJ众测平台" (Register now to join the TJ Zongce platform).



主页 简介 定价



## 简介

TJ众测是一个众包市场，允许个人和企业(称为Requester)使用24/7、全球分布的劳动力(称为Worker)执行任务。您可以在平台上法宝宝项目，如“识别水果篮图像中的红苹果”，“完成调查问卷”等。Worker可以使用这个网站查找要处理的任务、提交问题并管理他们的帐户。



主页 简介 定价



## 定价

您只需要为您创建的项目付钱。

您为平台上的每个项目支付的费用由两个部分组成:您支付给员工的金额，以及您根据支付给员工的金额支付给平台的费用。

## Worker Section

View or modify personal information.

Worker can view his personal information and make changes.

View account balance\Make a withdrawal and view the withdrawal history

Worker can view his account balance in this page, and he can make a withdrawal and view the withdrawal history.

TI众测

任务广场 我的任务 问题帮助 hyq 退出

基本信息

详细信息

我的钱包

真实姓名: hyq

性别: 男

昵称: 韩永琦

年龄: 17

手机号: 18117167110

邮箱: 1661752018@qq.com

经验值: 2

保存 取消

TI众测

任务广场 我的任务 问题帮助 hyq 退出

基本信息

详细信息

我的钱包

教育背景: 本科

专业: 软件工程

工作领域: 计算机

保存 取消

基本信息

详细信息

我的钱包

余额

提现

余额:

0 元

提现

基本信息

详细信息

我的钱包

余额

提现

日期	提现金额	提现方式
暂无数据		

[View Mission Square](#)

This part is for viewing the tasks, you can know about task creation time, bonus amount, level requirements, etc.

任务广场

按条件查找:		报酬:	时间:	选择日期	选择日期	关键字:	请输入任意关键字	Q 搜索
筛选:		<input checked="" type="checkbox"/> 显示可接受任务	<input checked="" type="checkbox"/> 显示等级过高任务	<input checked="" type="checkbox"/> 显示已完成任务	<input checked="" type="checkbox"/> 显示未完成任务			
排序:		热度	报酬	创建时间				
需求方	项目名称	热度	报酬	创建时间	行为			
34	选择问题	10000	50	preview	Quality	>		
34	对文化普及程度的调查	10000	40	preview	Quality	>		
34	对居住环境满意度的调查	10000	79	preview	Quality	>		
34	图片分类问题	10000	100	preview	Quality	>		
34	问卷调查	10000	10	undefined	preview	Accept		
34	问卷调查	10000	30	undefined	preview	Accept		
34	大学生对学校食堂满意度的情况问卷调查	10000	20	preview	Quality	>		

任务广场

按条件查找:		报酬:	时间:	选择日期	选择日期	关键字:	请输入任意关键字	Q 搜索
筛选:		<input checked="" type="checkbox"/> 显示可接受任务	<input checked="" type="checkbox"/> 显示等级过高任务	<input checked="" type="checkbox"/> 显示已完成任务	<input checked="" type="checkbox"/> 显示未完成任务			
排序:		热度	报酬	创建时间				
需求方	项目名称	热度	报酬	创建时间	行为			
34	选择问题	10000	50	preview	Quality	>		
Description: 问题均为二选一				Qualification 暂无	Level: 1 <span style="color: red;">✖ Not enough.</span>			
34	对文化普及程度的调查	10000	40	preview	Quality	>		
34	对居住环境满意度的调查	10000	79	preview	Quality	>		
34	图片分类问题	10000	100	preview	Quality	>		
34	问卷调查	10000	10	undefined	preview	Accept		
34	问卷调查	10000	30	undefined	preview	Accept		
34	大学生对学校食堂满意度的情况问卷调查	10000	20	preview	Quality	>		

You can search for tasks based on time, reward, keyword.

TI众测 广场 任务 帮助 hyq 登出

## 任务广场

按条件查找： 报酬： 时间： 关键字： 搜索 筛选： 显示可接受任务 显示等级过高任务 显示已完成任务 显示未完成任务 排序： 热度 报酬 创建时间

需求方	项目名称	热度	报酬	创建时间	行为	
34	图片分类问题	10000	100	preview	Quality	>
34	对居住环境满意度的调查	10000	79	preview	Quality	>
34	选择问题	10000	50	preview	Quality	>
34	对文化普及程度的调查	10000	40	preview	Quality	>
34	问卷调查	10000	30	undefined	preview	Accept
34	大学生对学校食堂满意度的情况问卷调查	10000	20	preview	Quality	>
34	问卷调查	10000	10	undefined	preview	Accept

TI众测 广场 任务 帮助 hyq 登出

## 任务广场

按条件查找： 报酬： 时间： 关键字： 搜索 筛选： 显示可接受任务 显示等级过高任务 显示已完成任务 显示未完成任务 排序： 热度 报酬 创建时间

需求方	项目名称	热度	报酬	创建时间	行为	
34	问卷调查	10000	10	undefined	preview	Accept
34	大学生对学校食堂满意度的情况问卷调查	10000	20	preview	Quality	>
34	问卷调查	10000	30	undefined	preview	Accept
34	对文化普及程度的调查	10000	40	preview	Quality	>
34	选择问题	10000	50	preview	Quality	>
34	对居住环境满意度的调查	10000	79	preview	Quality	>
34	图片分类问题	10000	100	preview	Quality	>

TI众测 广场 任务 帮助 hyq 登出

## 任务广场

按条件查找： 报酬： 时间： 关键字： 搜索 筛选： 显示可接受任务 显示等级过高任务 显示已完成任务 显示未完成任务 排序： 热度 报酬 创建时间

需求方	项目名称	热度	报酬	创建时间	行为	
34	选择问题	10000	50	preview	Quality	>
34	对文化普及程度的调查	10000	40	preview	Quality	>
34	对居住环境满意度的调查	10000	79	preview	Quality	>
34	图片分类问题	10000	100	preview	Quality	>
34	问卷调查	10000	10	undefined	preview	Accept
34	问卷调查	10000	30	undefined	preview	Accept
34	大学生对学校食堂满意度的情况问卷调查	10000	20	preview	Quality	>

## Learn about task details

You can learn about task details, you can just choose practice or accept the task, the system will judge whether it can accept this task according to your level.

The screenshot shows a dark-themed web interface. At the top left is the logo 'TI众测'. The top right has user information 'hyq' and a '登出' (Logout) button. The main content area has a teal header with the title '大学生服装知识调查' (College Students' Clothing Knowledge Survey). Below the title is a green button labeled '※ 进行中' (In Progress). Two dark blue buttons are below it: '测试' (Test) on the left and '练习' (Practice) on the right. A teal box contains project details: '项目类型: 问卷调查' (Project Type: Survey), '创建时间:' (Created Time:), '项目限制: 学历本科以上' (Project Limit: Undergraduate degree or above), '资质: Level 3' (Qualification: Level 3), '完成度: 100%' (Completion Rate: 100%), and '任务要求: 暂无' (Task Requirements: None).

[广场](#) > [大学生服装知识调查](#)

### 大学生服装知识调查

※ 进行中

测试

练习

项目类型: 问卷调查

创建时间:

项目限制: 学历本科以上

资质: Level 3

完成度: 100%

任务要求: 暂无

#### ① 项目描述

此调查需要答题人对服装知识有一定的了解，并对服饰做出评价

#### 目录

The screenshot shows a dark-themed web interface. At the top left is the logo 'TI众测'. The top right has user information 'hyq' and a '登出' (Logout) button. The main content area displays task details: '任务编号: 3', '任务名称: 大学生服装知识调查', '任务类型: 问卷调查', '发布者: boss', '发布日期: ', '完成度: 100%', '任务报酬: 10', and '任务描述:'. Below the task description is a note: '此调查需要答题人对服装知识有一定的了解，并对服饰做出评价'.

## Browse personal task list

Worker can view his task list in this page.

The screenshot shows a task management interface for workers. At the top, there are search filters for reward, time range, and keywords, along with a search button. Below that, there are filtering options for completed and uncompleted tasks. The main area displays a table of tasks:

需求方	项目描述	报酬	创建时间	完成度	备注
34	问卷调查	30	10%	<button>继续</button>	>
34	问卷调查	20	20%	<button>继续</button>	>
34	问卷调查	10	100%		

Pagination controls at the bottom indicate page 1 of 1.

## Get help

Worker can understand the system and get help in this page.

The screenshot shows a help section with a sidebar containing links to common questions, newbie guide, feedback, and personal settings. The main panel displays frequently asked questions:

- 什么是TJ众测?
- TJ众测主要解决什么问题?
- 使用TJ众测可以完成什么样的工作?

## Requester Section

#### View or modify personal information.

Requester can view his personal information and make changes.

个人信息

---

基本信息

---

邮箱 2333@qq.com

用户名 lyh

真实姓名 liuyue

性别 男 女

年龄 20

手机号码 13367403962

保存信息

个人信息

---

基本信息

详细信息

图片仓库

我的钱包

---

详细信息

---

公司/机构名称 tongji university

地址 上海

研究领域 computer science

保存信息

个人信息

- 基本信息
- 详细信息
- 图片仓库
- 我的钱包

## 图片仓库

First

Second

hello

abc

猫

狗

## 我的钱包

余额:

0 元

充值

View account balance\Recharge operation

Requester can view his account balance in this page, and he can do the recharge operation, in this part, he can also choose recharge method, is WeChat, Alipay or bank card.

个人信息

基本信息

详细信息

我的钱包

## 我的钱包

余额:

元

充值

提现

## Create Task

Requester can create task in this page.

创建项目

调查问卷

图片喜好

## 创建项目

借助任务模板来快捷地创建新项目，或创建自定义项目



调查问卷

提供一份调查问卷向参与者收集信息



图片识别

按照要求选择图片

You can fill in project properties, specify design layout and preview project.

项目属性

设计布局

请输入问卷

暂无数据

帮助 提交

## Modify or delete task

Requester can views tasks in this page.

管理项目

查看项目结果，对正在进行的项目部分设置进行修改

项目id	项目名称	创建时间	截止日期	项目状态	参与人数	操作
1	大学生对学校食堂满意度的情况问卷调查	2018-12-10 00:00:00	2018-12-12 00:00:00	0	0	<a href="#">查看结果</a>
2	大学生消费情况调查	2018-12-01 00:00:00	2018-12-13 00:00:00	0	1	<a href="#">查看结果</a>
8	知乎网的品牌知名度问卷调查	2019-01-20 13:00:00	2019-01-25 00:00:00	0	1	<a href="#">查看结果</a>
19	猫和狗的图片标记问题	2018-12-25 12:00:00	2018-12-27 12:00:00	0	1	<a href="#">查看结果</a>
20	药品市场调查问卷	2018-12-25 16:00:00	2018-12-27 09:00:00	0	0	<a href="#">查看结果</a>
21	两学一做知识竞赛问卷	2018-12-26 10:00:00	2018-12-28 07:00:00	0	0	<a href="#">查看结果</a>
22	“我看央视财经频道问卷”调查阿里巴巴	2018-12-26 10:00:00	2018-12-28 07:00:00	0	0	<a href="#">查看结果</a>
23	计算机专业调查	2018-12-24 00:00:00	2018-12-31 00:00:00	0	1	<a href="#">查看结果</a>

## View task response

Requester can view task response in this page.

TI众测

管理 创建 lyh 退出

答题情况分析

您所在的年级是? >

您的性别: >

您一个月中有多少天选择在学校食堂就餐? >

您认为学校食堂的饭菜价格如何? >

您平时喜欢去哪些食堂? >

您对食堂的饭菜分量如何评价? >

您觉得食堂的饭菜口味如何? >

您认为食堂服务人员的服务态度怎样? >

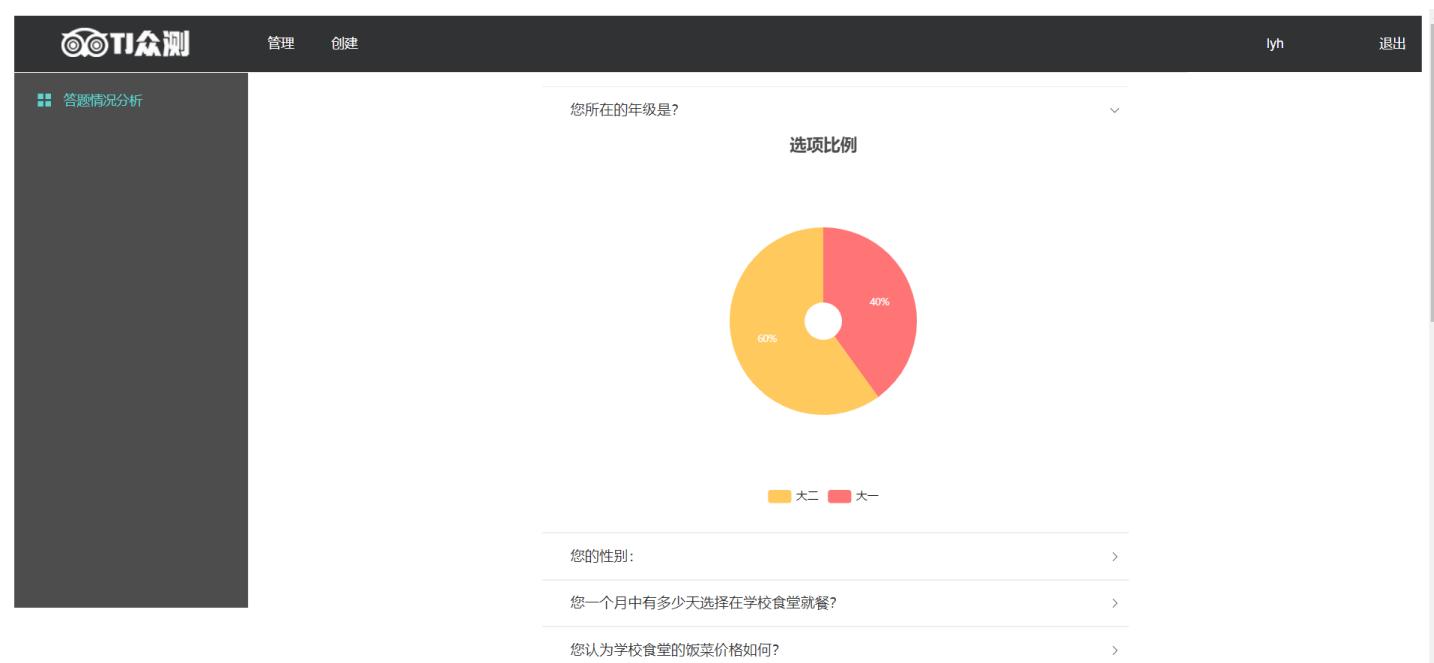
您在学校食堂的窗口经常遇到以下哪些情况? >

您觉得食堂早餐质量怎么样? >

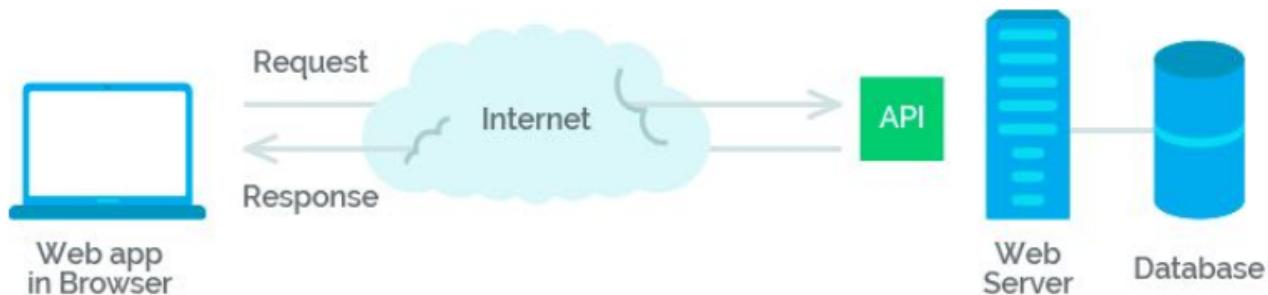
您希望食堂对那些方面进行改进? >

您认为食堂菜单更新情况如何? >

您认为食堂开放时间是否合理? >



## Business Architecture (业务架构)

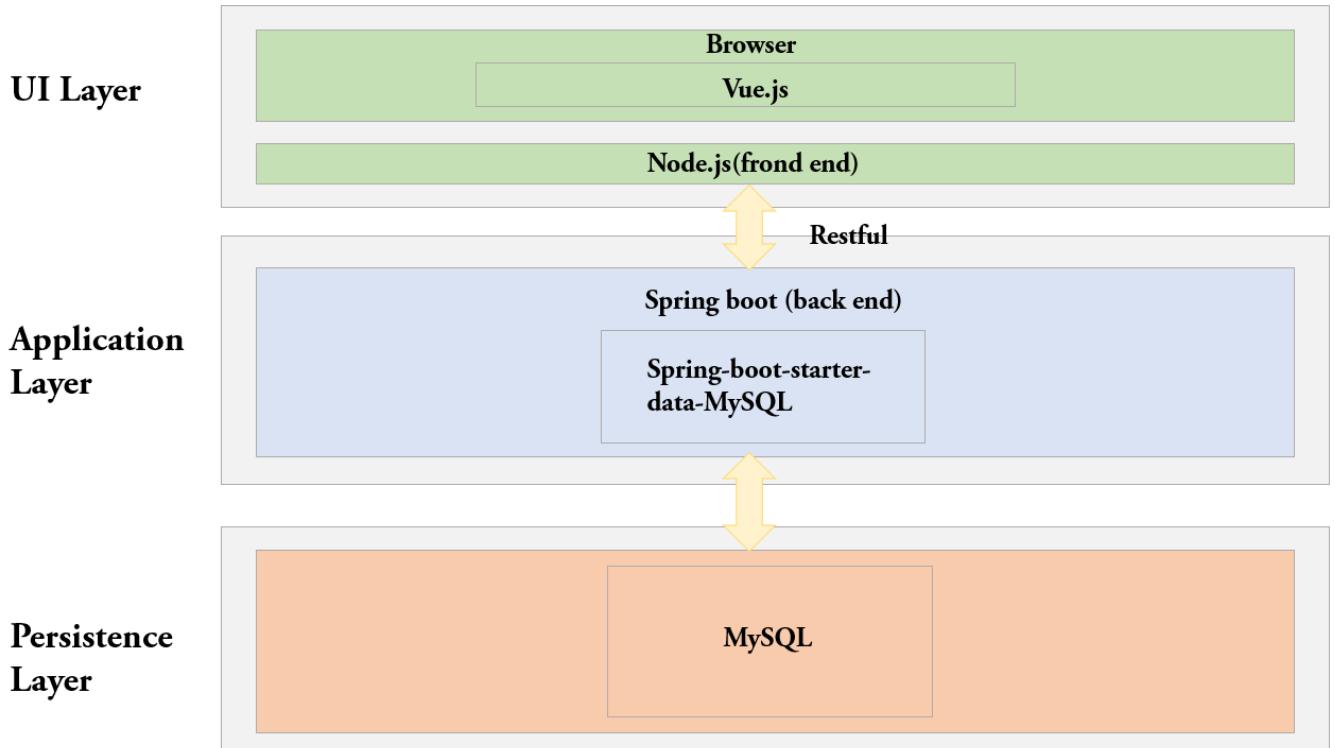


Restful is a web-based software architecture style designed to facilitate the transfer of information between different software/programs on a network such as the Internet.

First, the web app sends a request to the backend through the Internet in the browser. The back end sends data requests to the database through the code, and the back end returns the data to the front end, completing the complete separation of the front and back ends.

## Technology Architecture (技术架构)

The architecture of the technology stacks (coding language, frameworks, libraries, styles, etc) used in the project. Here is our Technology Architecture:

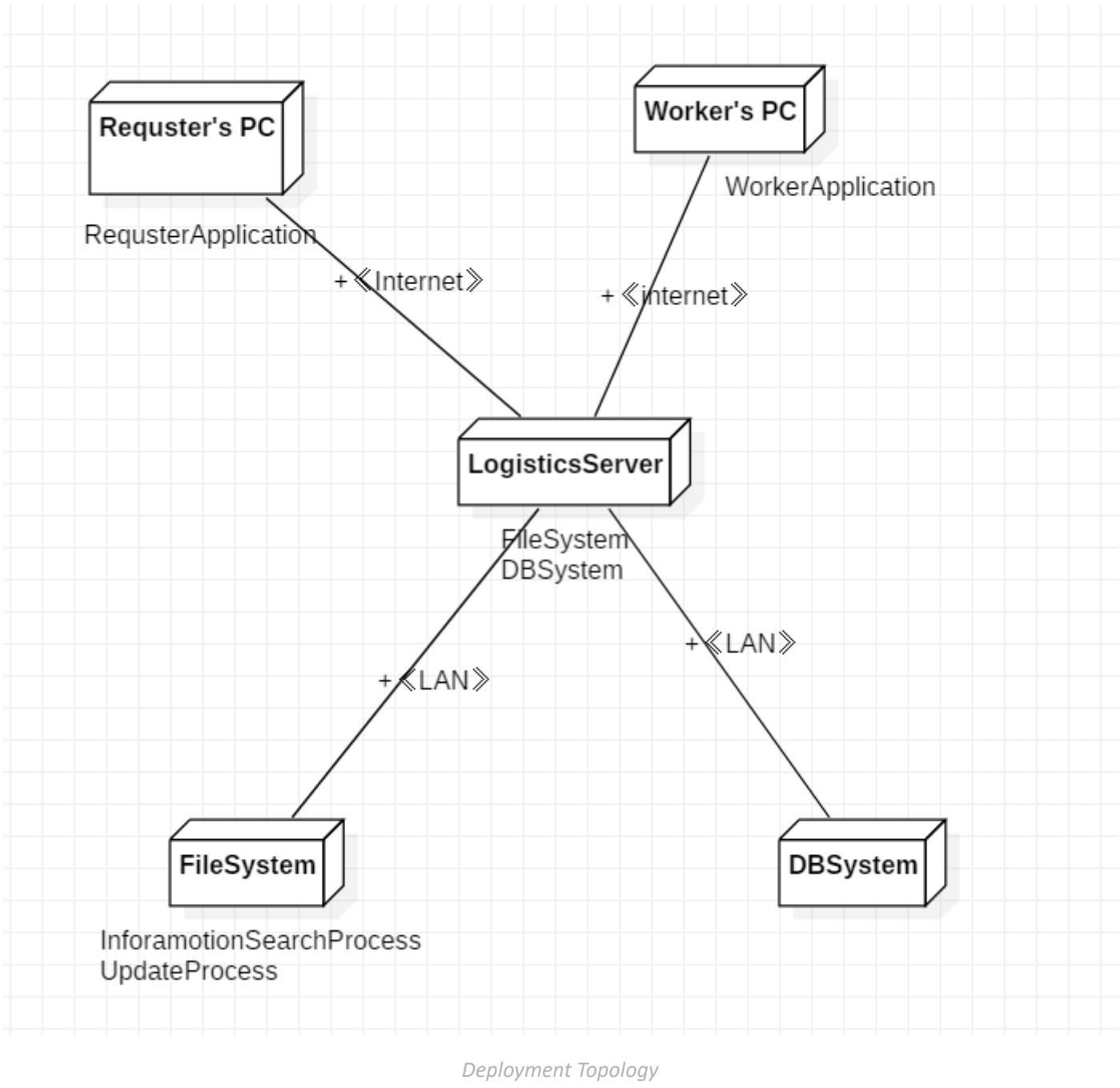


In the UI Layer, we use Vuejs as our front end frame and build the front end on the Nodejs.

In the Application Layer, we use Java as the primary development language for our backend. At the same time, we use the Spring-boot-starter-data-MySQL to Exchange data with the MySQL database.

In the Persistence Layer, we use MySQL for basic data storage, while using Redis to save cookies that record user login status. Spring-boot automatically caches logs and checks for errors or business records.

## Deployment Topology (部署结构)



*Deployment Topology*

Deployment Topology shows how the whole project is deployed in the physical environment, including servers, VMs/containers, networks, key components, etc. An overview picture is required to show the topology. Here is our Deployment Topology:

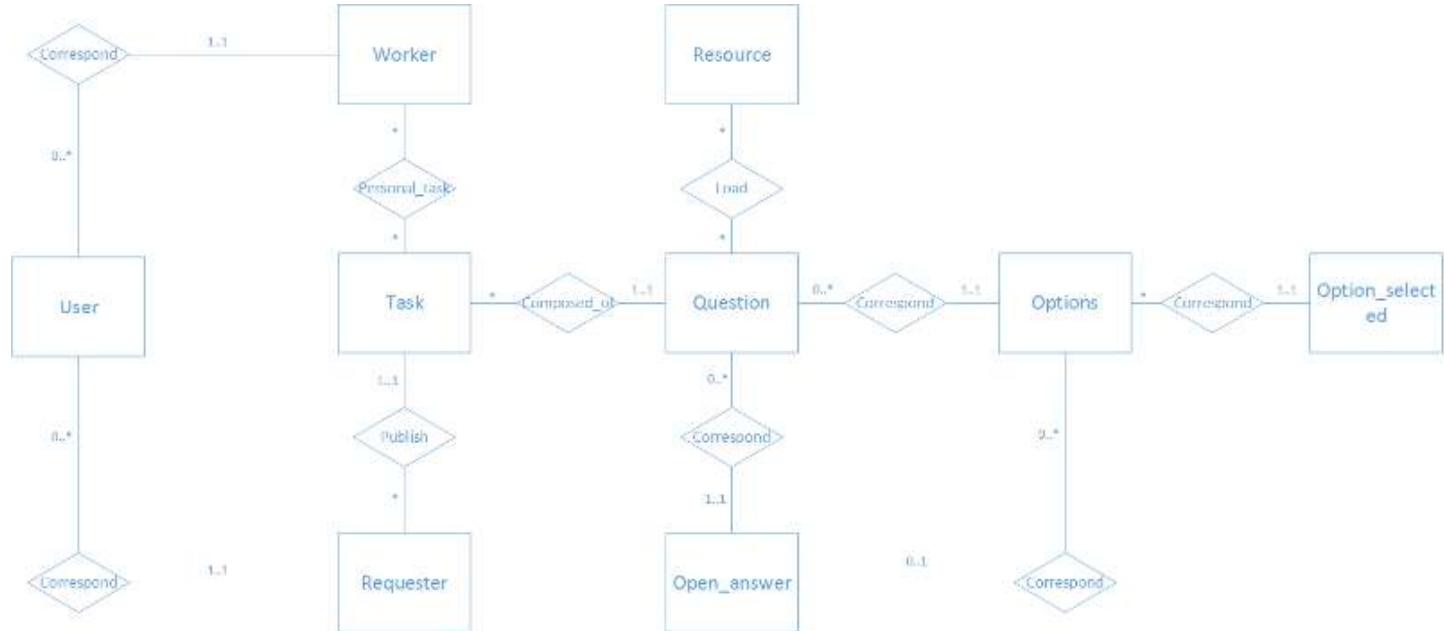
We use the method of front and rear separation to deploy, at the same time, the back end of our estimation system and the back end of the crowdsourcing system are respectively deployed on two computers, respectively interacting with different databases, but together for the same front-end service. This depends on spring's support for cross-domain access, and the low coupling between the estimate system and the crowdsourcing system.

## Data Models (数据建模)

### Entity-Relation Design (实体-关系设计)

An Entity-Relation Diagram is required to show the concepts in the project and their relations.

Here is our E-R diagram.



## Relationship

Worker-Task

Each worker can select multiple tasks to complete, and each task can be selected by multiple workers.

User-Worker

A user can choose to register as a worker, and each user\_id will uniquely correspond to a worker\_id.

Resource-Question

Each question can have multiple image resources, and each image resource can also correspond to multiple questions.

Task-Requester

Each requester can publish multiple tasks, each task uniquely corresponds to one publisher - requester

Question-Open\_answer

Each short answer question can have multiple answers, and each answer can only correspond to one question.

Question-Options

Each multiple-choice question can have multiple options, each option uniquely corresponding to a short answer question

Options- Option\_selected

Each option can have multiple choices of answers, and each answer can only uniquely correspond to one option.

## Database Design (数据库设计)

According to the Entity-Relationship Design, the detailed database design for entities and relations should be listed as tables. Here gives a table template:

**Table: User**

Column	Type	Description
username	varchar[30]	NOT NULL
password	varchar[10]	NOT NULL
role	varchar[10]	NOT NULL

**Table: Worker**

Column	Type	Description
worker_id	int unsigned	AUTO_INCREMENT
username	varchar[10]	NOT NULL
name	varchar[20]	NOT NULL
tele_number	varchar[11]	
e_mail	varchar[30]	
balance	int unsigned	
withdrawn_method	varchar[20]	
education	varchar[20]	
work_area	varchar[30]	
age	int unsigned	
level	int unsigned	
gender	int unsigned	
major	varchar[20]	

**Table: Requester**

Column	Type	Description
requester_id	int unsigned	AUTO_INCREMENT
username	varchar[10]	NOT NULL

Column	Type	Description
name	varchar[20]	NOT NULL
tele_number	varchar[11]	
e_mail	varchar[30]	
research_field	varchar[30]	
instituition_name	varchar[30]	
address	varchar[30]	
pay_method	varchar[20]	
gender	int signed	
age	int signed	
balance	int signed	

**Table: Task**

Column	Type	Description
task_id	int unsigned	AUTO_INCREMENT
name	varchar[50]	NOT NULL
decription	varchar[255]	NOT NULL
reward	int unsigned	
status	varchar[5]	
requester_id	int unsigned	NOT NULL
type	varchar[30]	
restrictions	varchar[30]	
start_time	datetime	
end_time	datetime	
population	int unsigned	
level	int unsigned	
time_limitation	int unsigned	
pay_time	int unsigned	

**Table: Question**

Column	Type	Description
question_id	int unsigned	NOT NULL
content	varchar[50]	NOT NULL
resource_loading	int unsigned	NOT NULL
type	int unsigned	NOT NULL
task_id	int unsigned	NOT NULL

**Table: Options**

Column	Type	Description
option_id	int unsigned	NOT NULL
content	varchar[20]	NOT NULL
question_id	int unsigned	NOT NULL
open_answer_permission	int unsigned	NOT NULL
option_number	int unsigned	NOT NULL

**Table: Option\_selected**

Column	Type	Description
option_selected_id	int unsigned	NOT NULL
option_id	int unsigned	NOT NULL
worker_id	int unsigned	NOT NULL
time	datetime	NOT NULL

**Table: Open\_answer**

Column	Type	Description
open_answer_id	int unsigned	NOT NULL
content	varchar[50]	NOT NULL

Column	Type	Description
question_id	int unsigned	NOT NULL
worker_id	int unsigned	NOT NULL
time	datetime	NOT NULL

Table: Resource

Column	Type	Description
resource_id	int unsigned	NOT NULL
link	varchar[50]	NOT NULL
type	varchar[10]	NOT NULL

Table: QuestionResource

Column	Type	Description
question_id	int unsigned	NOT NULL
resource_id	int unsigned	NOT NULL

Table: Personal\_task

Column	Type	Description
worker_id	int unsigned	NOT NULL
task_id	int unsigned	NOT NULL

## Interface Specifications (接口规约)

The specifications of interfaces used in interactions among subsystems

### Worker

## workerUpdate

### Description (接口描述)

This interface is used to update one worker's information.

Request Method	POST
Request Address	<a href="http://localhost:8080/personal-task/add">http://localhost:8080/personal-task/add</a>

### Header (参数)

Name	in	Located Description	Required	Example
X_Auth_Token	string	the token of user	Yes	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiIxNjYxNzUyMDE4QHFxLmNvbSlslmN

### Responses (返回结果)

Code	Message
201	Succeed

### Request Sample (示例请求)

```
worker/update
```

### Response Sample (示例结果)

```
{
  "code": "201",
  "message": "succeed"
}
```

## workerDelete

### Description (接口描述)

This interface is used to delete data in database identifying by id of the worker.

Request Method	DELETE
Request Address	<a href="http://localhost:8080/worker/delete">http://localhost:8080/worker/delete</a>

## personalTaskAdd

### Description (接口描述)

This interface is used to add pairs of workers and tasks. Besides, this interface is temporary. After the natural join logic finishes, this interface will update.

Request Method	POST
Request Address	<a href="http://localhost:8080/personal-task/add">http://localhost:8080/personal-task/add</a>

### Header (参数)

Located				
Name	in	Description	Required	Example
X_Auth_Token	string	the token of user	Yes	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjEjw

### Body (参数)

Name	Located in	Description	Required
taskId		integer	The id of the task

### Responses (返回结果)

Code	Message
201	Succeed

### Request Sample (示例请求)

personal-task/add

## Response Sample (示例结果)

```
{  
  "code": "201",  
  "message": "succeed"  
}
```

## findMyself

### Description (接口描述)

This interface is used to check the user's(worker) information.

Request Method	GET
Request Address	<a href="http://localhost:8080/worker/find-myself">http://localhost:8080/worker/find-myself</a>

### Header (参数)

Located					
Name	in	Description	Required	Example	
X_Auth_Token	string	the token of user	Yes	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjE jw	

### Responses (返回结果)

Code	Username	schema
200	good_day	worker:object

### Request Sample (示例请求)

worker/find-myself

## Response Sample (示例结果)

```
{  
  "code": "200",  
  - "worker": {  
    "username": "good_day",  
    "name": "happy_all_the_time",  
    "teleNumber": 0,  
    "balance": 0,  
    "withdrawnMethod": null,  
    "education": null,  
    "workArea": null,  
    "age": 0,  
    "workerId": 16,  
    "email": null  
  }  
}
```

## findMyTask

### Description (接口描述)

This interface is used to get my(worker) task lists.

Request Method	GET
Request Address	<a href="http://localhost:8080/personal-task/find-my-task">http://localhost:8080/personal-task/find-my-task</a>

### Header (参数)

Located				
Name	in	Description	Required	Example
X_Auth_Token	string	the token of user	Yes	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjEjw

### Responses (返回结果)

Code	Name	schema
200	experiment	tasks:list<object>

## Request Sample (示例请求)

```
personal-task/find-my-task
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "tasks": [
    {
      "name": "experiment",
      "description": "this is a work needing to collect results", "reward": 100,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 1,
      "task_id": 2
    }
  ]
}
```

## selectOne

### Description (接口描述)

This interface is used to select one option.

Request Method	POST
Request Address	<a href="http://localhost:8080/question/select-one">http://localhost:8080/question/select-one</a>

### Header (参数)

Located					
Name	in	Description	Required	Example	
X_Auth_Token	string	the token of user	Yes	yJhbGciOiJIUzUxMiJ9eyJzdWliOiJzdW5zaGluZSIsImNyZWF0ZWQiOjE1	

## Body (参数)

Name	Located in	Description	Required
optionId	integer	the id of option	Yes

## Responses (返回结果)

Code	Message
200	succeed

## Request Sample (示例请求)

```
question/select-one
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "message": "succeed"
}
```

## answerOne

### Description (接口描述)

This interface is used to answer a question.

Request Method	POST
Request Address	<a href="http://localhost:8080/question/answer-one">http://localhost:8080/question/answer-one</a>

## Header (参数)

Name	in	Located	Description	Required	Example

Located				
Name	in	Description	Required	Example
X_Auth_Token	string	the token of user	Yes	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjE

## Body (参数)

Name	Located in	Description	Required
content	string	the content of the answer	Yes
optionId	integer	the id of the option	Yes

## Responses (返回结果)

Code	Message
200	succeed

## Request Sample (示例请求)

```
question/answer-one
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "message": "succeed"
}
```

## seeMyQuestion

### Description (接口描述)

This interface is used to see my answer of a question.

Request Method	GET
----------------	-----

Request Address

<http://localhost:8080/question/see-my-answer>

## Header (参数)

Located				
Name	in	Description	Required	Example
X_Auth_Token	string	the token of user	Yes	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjE

## Body (参数)

Name	Located in	Description	Required	Example
taskId	integer	the id of the task	Yes	6

## Responses (返回结果)

Code	Message	schema
200	success	questions:list<object>

## Request Sample (示例请求)

```
question/see-my-answer
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "Questions": [
    { "question": {
        "id": 3,
        "content": "问卷调查",
        "resource_loading": 0,
        "type": 0,
        "task_id": 6
      },
      "options": [
        ...
      ]
    }
  ]
}
```

```
{  
  "id": 1,  
  "content": "是",  
  "questionId": 3,  
  "openAnswerPermittion": 0,  
  "optionNumber": 1  
},  
{  
  "id": 2,  
  "content": "随便写",  
  "questionId": 3,  
  "openAnswerPermittion": 1,  
  "optionNumber": 2  
}  
],  
"resources": null,  
"selecteds": [  
  { "id": 1,  
    "optionId": 1,  
    "workerId": 22,  
    "time": "2018-12-04T08:08:37.000+0000"  
  }  
],  
"answers": [  
  { "id": 1,  
    "content": "没什么想法",  
    "questionId": 3,  
    "workerId": 22,  
    "time": "2018-12-04T08:08:41.000+0000",  
    "optionId": 2  
  }  
]  
}  
]  
}
```

## \*\*Tourist \*\*

### *taskFindAll*

#### Description (接口描述)

This interface is used to get all the tasks which are arranged in reverse order.

Request Method

GET

Request Address

<http://localhost:8080/task/find-all>

## Responses (返回结果)

Code	Message	schema
200	success	tasks:list<object>

## Request Sample (示例请求)

```
task/find-all
```

## Response Sample (示例结果)

```
{
  "code": "200",
  -"tasks": [
    -{
      "name": "dsad",
      "description": "fdfd",
      "reward": 30,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 1,
      "task_id": 9
    },
    -{
      "name": "big_data",
      "description": "this is a task to distinguish",
      "reward": 40,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 1,
      "task_id": 8
    },
    -{
      "name": "classify",
      "description": "this is a task to classify",
    }
  ]
}
```

```
"reward": 15,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 2,
"task_id": 7
},
-{
"name": "Tag",
"description": "this is a tagging task",
"reward": 30,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 6
},
-{
"name": "math",
"description": "this is a problem to solve",
"reward": 5,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 3
},
-{
"name": "experiment",
"description": "this is a work needing to collect results",
"reward": 100,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 2
},
-{
"name": "study",
"description": "this is a study task",
"reward": 25,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
```

```

"requester_id": 2,
"task_id": 1
}
]
}

```

## workerAdd

### Description (接口描述)

This interface is used to add rows to the table "Worker". Because of "not null" restrictions, parameter "username" and "name" must be input.

Request Method

POST

Request Address

<http://localhost:8880/worker/add>

### Body (参数)

Name	Located in	Description	Required	Example
username	string	Username of the worker	Yes	lyh
name	string	Username of the worker	Yes	liuyuehan
teleNumber	string	teleNumber of the worker	Yes	13956642750
eMail	string	eMail of the worker	Yes	<a href="mailto:1661752019@qq.com">1661752019@qq.com</a>
withdrawnMethod	string	withdrawnMethod of the worker	Yes	支付宝
education	string	education situation of the worker	Yes	university
workArea	string	workArea of the worker	Yes	software engineering
age	string	age of the worker	Yes	19
gender	string	gender of the worker	Yes	0
major	string	major of the user	Yes	

### Responses (返回结果)

WorkerID	code	Message
27	201	succeed

## Request Sample (示例请求)

```
worker/add
```

## Response Sample (示例结果)

```
{
  "workerId": 27,
  "code": "201",
  "message": "succeed"
}
```

## requesterAdd

### Description (接口描述)

This interface is used to add rows to the table "Requester". Because of "not null" restrictions, parameter "username" and "name" must be input.

Request Method	POST
Request Address	<a href="http://localhost:8080/requester/add">http://localhost:8080/requester/add</a>

### Body (参数)

Name	Located in	Description	Required	Example
username	string	The username of the requester	Yes	lyh
name	string	The name of the requester	Yes	liuyue
teleNumber	string	the teleNumber of the requester	Yes	13367403962
eMail	string	eMail of the requester	Yes	<a href="mailto:1661752019@qq.com">1661752019@qq.com</a>
research_field	string	research_field of the requester	Yes	medical
institutionName	string	institutionName of the requester	Yes	tongji university
address	string	address of the requester	Yes	上海
payMethod	string	payMethod of the requester	Yes	支付宝

Name	Located in	Description	Required	Example
gender	string	gender of the requester	Yes	0
age	string	age of the requester	Yes	30

## Responses (返回结果)

WorkerID	Message
201	succeed

## Request Sample (示例请求)

```
requester/add
```

## Response Sample (示例结果)

```
{
  "code": "201",
  "message": "succeed"
}
```

## register

### Description (接口描述)

This interface is used to register. The return value depends on the situation of adding.

Request Method	POST
Request Address	<a href="http://localhost:8080/register">http://localhost:8080/register</a>

### Body (参数)

Name	Located in	Description	Required	Example
username	string	the email of the user	Yes	<a href="mailto:1661752018@qq.com">1661752018@qq.com</a>

Name	Located in	Description	Required	Example
name	string	the password of user	Yes	tongji
role	string	the role of user	Yes	ROLE_WORKER

## Responses (返回结果)

WorkerID	Message
201	succeed

## Request Sample (示例请求)

```
register
```

## Response Sample (示例结果)

```
{
  "code": "201",
  "message": "succeed"
}
```

## login

### Description (接口描述)

This interface is used to login, it will return with a token.

Request Method	POST
Request Address	<a href="http://localhost:8080/login">http://localhost:8080/login</a>

### Body (参数)

Name	Located in	Description	Required	Example
username	string	the email of the user	Yes	<a href="mailto:1661752018@qq.com">1661752018@qq.com</a>

Name	Located in	Description	Required	Example
name	string	the password of user	Yes	tongji
role	string	the role of user	Yes	ROLE_WORKER

## Responses (返回结果)

WorkerID X\_Auth\_Token

201	eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJnb29kX2RheSIsImNyZWF0ZWQiOjE1NDIxNjgzNTk0MDgsImV4cCl6MTU0Mjc3I_p3qMWYcq0cnX9cllWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vq6SXOF8lwX7zPngYE8ha2ygxQ
-----	---

## Request Sample (示例请求)

login

## Response Sample (示例结果)

```
{
  "code": "200",
  "X_Auth_Token": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJnb29kX2RheSIsImNyZWF0ZWQiOjE1NDIxNjgzNTk0MDgsImV4cCl6MTU0Mjc3I_p3qMWYcq0cnX9cllWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vq6SXOF8lwX7zPngYE8ha2ygxQ",
  "message": ""
}
```

# Requester

*taskAdd*

## Description (接口描述)

This interface is used to add a new task.

Name,description and requester\_id is must.

This can only be used by requester.

Request Method

POST

Request Address

<http://localhost:8880/task/add>

### Header (参数)

Name	Located in	Description	Required	Example
X_Auth_Token		string	the token of user	Yes

### Body (参数)

Name	Located in	Description	Required	Example
name	string	the name of the task	Yes	Tag
description	string	the description of the task	Yes	this task require some people to do tagging
reward	integer	the reward of task	Yes	20
status	string	the status of the task	Yes	良好
type	string	the type of the task	Yes	问卷调查
restrictions	string	the restrictions of the task	Yes	学生
start_time	datetime	the start_time of the task	Yes	2018-12-10 00:00:00
end_time	datetime	the end_time of the task	Yes	2018-12-10 00:00:00
level	integer	the level of the task	Yes	1
time_limitation	integer	the time_limitation of the task	Yes	3
pay_time	integer	the pay_time of the task	Yes	1

### Responses (返回结果)

Code	TaskID	Message
201	14	succeed

### Request Sample (示例请求)

```
task/add
```

## Response Sample (示例结果)

```
{  
    "code": "201",  
    "message": "succeed",  
    "taskId": 14  
}
```

## taskUpdate

### Description (接口描述)

The interface is used to update a task.

This can only be used by requester.

Request Method	PUT
Request Address	<a href="http://localhost:8080/task/update">http://localhost:8080/task/update</a>

### Header (参数)

Located				
Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiIyMzMzQHFxLmNvbSIsImNyZWFOZVFzB7pad8pVoyz8maOVzSxVxLNrip7lVA

## taskDataAdd

### Description (接口描述)

The interface is used to add a task's history.

The number of worker have get involved in the task and the rate of the task have been completed are needed.

Request Method	POST
Request Address	<a href="http://localhost:8080/taskData/add">http://localhost:8080/taskData/add</a>

### Header (参数)

Located				
Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJnb29kX2RheSlsImNyZWF0ZWQiOjE_p3qMWYcq0cnX9clIWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vq6

## Body (参数)

Name	Located in	Required	Description	Example
id	integer	是	the id of task	1
worker_num	integer	是	the number of worker have get involved in the task	50
progress_rate	float	是	the rate of the task have been completed	0.8

## Responses (返回结果)

Code	Message
201	succeed

## Request Sample (示例请求)

```
taskData/add
```

## Response Sample (示例结果)

```
{
  "code": "201",
  "message": "succeed",
}
```

## requesterUpdate

### Description (接口描述)

This interface is used to update one worker's information.

Request Method	PUT			
----------------	-----	--	--	--

Request Address <http://localhost:8880/requester/update>

## Header (参数)

Located				
Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOilyMzMzQHFxLmNvbSlsImNyZWFOZvFazB7pad8pVoyz8maOVzSxVxLNrip7lVA

## *requesterDelete*

### Description (接口描述)

This interface is used to delete data in the table "Requester" identified by id.

Request Method	DELETE			
----------------	--------	--	--	--

Request Address <http://localhost:8080/requester/delete>

## *findMyself*

### Description (接口描述)

The interface is used to get user's(requester) information.

Request Method	GET			
----------------	-----	--	--	--

Request Address <http://localhost:8080/requester/find-myself>

## Header (参数)

Located				
Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOjoiYXBweV9kYXkiLCJjcmVhdGVkljoxNmJ_AVofkYdrom3z4Kf47Uw_4cs8_hgG1N34rxxuS-GNFB5_265xIKCJAfn

## Responses (返回结果)

Code	Message	Schema
201	succeed	requester:object

## Request Sample (示例请求)

```
requester/find-myself
```

## Response Sample (示例结果)

```
{
  "requester": {
    "requesterId": 21,
    "username": "sunshine",
    "name": "happy_all_the_time",
    "teleNumber": 0,
    "area": null,
    "institutionName": null,
    "payMethod": null,
    "email": null
  },
  "code": "200"
}
```

## questionAdd

### Description (接口描述)

This interface is used to add question.

Request Method	POST
Request Address	<a href="http://localhost:8880/question/add-question">http://localhost:8880/question/add-question</a>

### Header (参数)

Located				
Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJzdW5zaGluZSIsImNyZWF0ZWQiOjE

## Body (参数)

Name	Located in	Required	Description	Example
taskId	integer	是	the id of task	1
content	string	是	the content of question	问卷调查
resourceLoading	integer	是		0
type	integer	是	the type of the question	0

## Responses (返回结果)

Code	Message	Schema
201	succeed	questionID:object

## Request Sample (示例请求)

```
question/add-question
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "questionId": 4,
  "message": "succeed"
}
```

## optionAdd

### Description (接口描述)

This interface is used to add option to a question.

Request Method	POST
Request Address	<a href="http://localhost:8880/question/add-option">http://localhost:8880/question/add-option</a>

## Header (参数)

Name	in	Located Required
X_Auth_Tokenstring	是	the token of user

## Body (参数)

Name	Located in	Required	Description	Example
content	string	是	the content of the option	是
questionId	integer	是	the id of the question	1
openAnswerPermission	integer	是		
optionNumber	integer	是	the number of the option	1

## Responses (返回结果)

Code	Message	Schema
201	succeed	optionID:object

## Request Sample (示例请求)

question/add-option
---------------------

## Response Sample (示例结果)

```
{
  "code": "200",
  "optionId": 3,
  "message": "succeed"
}
```

## putImage

### Description (接口描述)

This interface is used to put image.

Request Method	POST
----------------	------

| Request Address | <http://localhost:8080/put-image> |

### Header (参数)

Name	in	Located	Required
X_Auth_Tokenstring	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWJlOiJzdW5zaGluZSIsImNyZWF0ZWQiOjE1NDIxNzE

### Body (参数)

Name	Located in	Required	Description	Example
multipartFile	file	是	the picture	
questionId	integer	是	the id of the question	6
type	string	是	the type of the resource	

### Responses (返回结果)

Code	Message	Schema
201	succeed	file_name:object

## Request Sample (示例请求)

```
put-image
```

## Response Sample (示例结果)

```
{  
  "code": "200",  
  "file_name": "c4ba4d95-12da-466b-a55c-ba1c04a38b0b.png" }
```

## taskAttribute

### Description (接口描述)

Request Method	POST
Request Address	<a href="http://localhost:8080/task/attribute">http://localhost:8080/task/attribute</a>

### Header (参数)

Name	in	Required	Located	Example
			Description	
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJzdW5zaGluZSIsImNyZWF0ZWQiOjE

### Body (参数)

Name	Located in	Required	Description	Example
content	string	是	the content of the option	是
questionId	integer	是	the id of the question	1
openAnswerPermission	integer	是		
optionNumber	integer	是	the number of the option	1

### Responses (返回结果)

Code	Message	Schema
201	succeed	optionID:object

## Request Sample (示例请求)

question/add-option

## Response Sample (示例结果)

```
{
  "code": "200",
  "optionId": 3,
  "message": "succeed"
}
```

## putImage

### Description (接口描述)

This interface is used to put image.

Request Method	POST
Request Address	<a href="http://localhost:8080/put-image">http://localhost:8080/put-image</a>

### Body (参数)

Name	Located in	Required	Description	Example
id	string	是		
name	string	是		
describe	string	是		
tag	string	是		
reward	string	是	项目报酬	

Name	Located in	Required	Description	Example
population	string	是	参与人数	
finish_time	string	是	完成项目的时限	
limitation	string	是	项目的起止时间	
pay_time	string	是	自动支付的时间	
condition	string	是		
if_expert	string	是		

## General

### taskFindByld

#### Description (接口描述)

This interface is used to find a task by its id.

Request Method	GET
Request Address	<a href="http://localhost:8080/task/find-by-id">http://localhost:8080/task/find-by-id</a>

#### Query (参数)

Name	Located in	Required	Description	Example
X_Auth_Token	integer	是	the id of task to be find	6

#### Responses (返回结果)

Code	Message	Schema
201	succeed	requiemt:object

#### Request Sample (示例请求)

task/find-by-id
-----------------

## Response Sample (示例结果)

```
{  
  "name": "Tag",  
  "description": "this is a tagging task",  
  "reward": 30,  
  "status": null,  
  "type": null,  
  "restrictions": null,  
  "requests": null,  
  "requester_id": 1,  
  "task_id": 6  
}
```

## taskFindByName

### Description (接口描述)

The interface is used to find tasks by name.

Request Method	GET
Request Address	<a href="http://localhost:8080/task/find-by-name">http://localhost:8080/task/find-by-name</a>

### Header (参数)

Name	in	Required	Located	Description	Example
X_Auth_Token	string	是		the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjE_p3qMWYcq0cnX9clIWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vq6

### Query (参数)

Name	Located in	Required	Description	Example
name	string	是	the name of task to be found	classify

### Responses (返回结果)

Code	Message	Schema
201	succeed	tasks:object

## Request Sample (示例请求)

```
task/find-by-name
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "tasks": [
    { "name": "classify",
      "description": "this is a task to classify",
      "reward": 15,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 2,
      "task_id": 7
    }
  ]
}
```

## taskFindByRequesterId

### Description (接口描述)

The interface is used to find tasks by name.

Request Method	GET
Request Address	<a href="http://localhost:8080/task/find-by-requester-id">http://localhost:8080/task/find-by-requester-id</a>

### Header (参数)

Name	in	Required	Located	Description	Example

Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjE_p3qMWYcq0cnX9clIWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vq6

## Query (参数)

Name	Located in	Required	Description	Example
requesterid	integer	是	requester's id whose tasks need to be found	1

## Responses (返回结果)

Code	Message	Schema
201	succeed	tasks:list<object>

## Request Sample (示例请求)

```
task/find-by-requester-id
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "tasks": [
    {
      "name": "dsad",
      "description": "fdfd",
      "reward": 30,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 1,
      "task_id": 9
    },
    {
      "name": "big_data",
      "description": "this is a task to distinguish",
      "reward": 40,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 1
    }
  ]
}
```

```

"task_id": 8
},
{ "name": "Tag",
"description": "this is a tagging task",
"reward": 30,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 6
},
{ "name": "math",
"description": "this is a problem to solve", "reward": 5,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 3
},
{ "name": "experiment",
"description": "this is a work needing to collect results",
"reward": 100,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 2
}
]
}

```

## \*taskFindByReward \*

### Description (接口描述)

The interface is used to find tasks whose reward is in the range of user's expectation.

Request Method	GET
Request Address	<a href="http://localhost:8080/task/find-by-reward">http://localhost:8080/task/find-by-reward</a>

### Header (参数)

Located				
Name	in	Required	Description	Example
X_Auth_Token	string	是	the token of user	eyJhbGciOiJIUzUxMiJ9eyJzdWliOiJnb29kX2RheSlsImNyZWF0ZWQiOjE_p3qMWYcq0cnX9clIWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vq6

## Query (参数)

Name	Located in	Required	Description	Example
lowest	integer	是	the lowest reward of user's expectation	20
highest	integer	是	the highest reward of user's expectation	50

## Responses (返回结果)

Code	Message	Schema
201	succeed	tasks:list<object>

## Request Sample (示例请求)

```
task/find-by-reward
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "tasks": [
    {
      "name": "study",
      "description": "this is a study task",
      "reward": 25,
      "status": null,
      "type": null,
      "restrictions": null,
      "requests": null,
      "requester_id": 2,
      "task_id": 1
    },
    {
      "name": "Tag",
      "description": "this is a Tag task"
    }
  ]
}
```

```

"description": "this is a tagging task",
"reward": 30,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 6
},
{
"name": "big_data",
"description": "this is a task to distinguish", "reward": 40,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 8
},
{
"name": "dsad",
"description": "fdfd",
"reward": 30,
"status": null,
"type": null,
"restrictions": null,
"requests": null,
"requester_id": 1,
"task_id": 9
},
]
}

```

## *taskDelete*

### Description (接口描述)

The interface is used to delete a task, the id of it required to be provided.

Request Method	DELETE
Request Address	<a href="http://localhost:8080/task/delete">http://localhost:8080/task/delete</a>

## *taskDataFind*

## Description (接口描述)

The interface is used to find a task's history.

Request Method	GET
Request Address	<a href="http://localhost:8080/taskData/find">http://localhost:8080/taskData/find</a>

## Header (参数)

Name	in	Required	Located	Description	Example
X_Auth_Token	string	是	the token of user	the token of user	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJnb29kX2RheSIsImNyZWF0ZWQiOjE_p3qMWYcq0cnX9cllWGmj6QXS9T8Ss694PaGqPDcee5_rK1JeNp5_vqf

## Query (参数)

Name	Located in	Required	Description	Example
id	integer	是	the id of task whose information need to be found	6

## Responses (返回结果)

Code	Message	Schema
201	succeed	taskData:list<object>

## Request Sample (示例请求)

```
taskData/find
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "taskData": [
    { "prime": {
      { "id": 1,
        "time": "2018-11-14T05:07:52.000+0000"
      }
    }
  ]
}
```

```

},
"worker_num": 50,
"progress_rate": 0.8,
"id": 1,
"time": "2018-11-14T05:07:52.000+0000"
}
]
}

```

## *taskDelete*

### Description (接口描述)

The interface is used to delete a task, the id of it required to be provided.

Request Method	DELETE
Request Address	<a href="http://localhost:8080/task/delete">http://localhost:8080/task/delete</a>

## *workerFindById*

### Description (接口描述)

This interface is used to find worker's information through worker' ID.

Request Method	GET
Request Address	<a href="http://localhost:8080/worker/find-by-id">http://localhost:8080/worker/find-by-id</a>

### Query (参数)

Name	Located in	Required	Description	Example
id	integer	是	The id of the worker(primary key)	1

### Responses (返回结果)

Code	Message	Schema
201	succeed	worker:object

## Request Sample (示例请求)

```
worker/find-by-id
```

## Response Sample (示例结果)

```
{
  "code": "200",
  - "worker": {
    "username": "qsw",
    "name": "qiansiwei",
    "teleNumber": 123456789,
    "balance": 50,
    "withdrawnMethod": "paypal",
    "education": "college",
    "workArea": "software_engineering",
    "age": 1,
    "workerId": 1,
    "email": "qiansiwei@tongji.edu.cn"
  }
}
```

## workerFindByUsername

### Description (接口描述)

This interface is used to find worker's information through his username. Usually, this interface is used after a worker log in the system.

Request Method	GET
Request Address	<a href="http://localhost:8080/worker/find-by-username">http://localhost:8080/worker/find-by-username</a>

### Query (参数)

Name	Located in	Required	Description	Example
username	string	是	The username of worker	qsw

### Header (参数)

Name	Located in	Required	Description	Example
username	string	是		The token of user

## Responses (返回结果)

Code	Message	Schema
201	succeed	worker:object

## Request Sample (示例请求)

```
worker/find-by-username
```

## Response Sample (示例结果)

```
{
  "code": "200",
  - "worker": {
    "username": "qsw",
    "name": "qiansiwei",
    "teleNumber": 123456789,
    "balance": 50,
    "withdrawnMethod": "paypal",
    "education": "college",
    "workArea": "software_engineering",
    "age": 1,
    "workerId": 1,
    "email": "qiansiwei@tongji.edu.cn"
  }
}
```

## requesterFindByid

### Description (接口描述)

This interface is used to find requester's information through id.

Request Method	GET
Request Address	<a href="http://localhost:8080/requester/find-by-id">http://localhost:8080/requester/find-by-id</a>

## Query (参数)

Name	Located in	Required	Description	Example
id	integer	是	The id of the requester	1

## Responses (返回结果)

Code	Message	Schema
201	succeed	requester:object

## Request Sample (示例请求)

```
requester/find-by-id
```

## Response Sample (示例结果)

```
{
  "requester": {
    "requesterId": 1,
    "username": "hyq",
    "name": "hyq",
    "teleNumber": 12345,
    "area": "harbin",
    "institutionName": null,
    "payMethod": "paypal",
    "email": "405579949@qq.com"
  },
  "code": "200"
}
```

## requesterFindByUsername

### Description (接口描述)

This interface is used to find requester's information through his username. Usually, this interface is used after a requester log in.

Request Method	GET
----------------	-----

Request Address

<http://localhost:8080/requester/find-by-username>

## Query (参数)

Name	Located in	Required	Description	Example
username	string	是	The username of the requester	hyq

## Header (参数)

Name	Located in	Required	Description	Example
username	string	是	The token of user	

## Responses (返回结果)

Code	Message	Schema
201	succeed	requester:object

## Request Sample (示例请求)

```
requester/find-by-username
```

## Response Sample (示例结果)

```
{
  "requester": {
    "requesterId": 1,
    "username": "hyq",
    "name": "hyq",
    "teleNumber": 12345,
    "area": "harbin",
    "institutionName": null,
    "payMethod": "paypal",
    "email": "405579949@qq.com"
  },
  "code": "200"
}
```

## personalTaskFindTaskList

### Description (接口描述)

This interface is used to find a list of task's ids by worker's id which the worker joined in.

Request Method	GET
Request Address	<a href="http://localhost:8080/personal-task/find-task-list">http://localhost:8080/personal-task/find-task-list</a>

### Query (参数)

Name	Located in	Required	Description	Example
id	integer	是	The id of the worker	1

### Header (参数)

Name	Located in	Required	Description	Example
username	string	是	The token of user	

### Responses (返回结果)

Code	Message	Schema
201	succeed	requiement:object

### Request Sample (示例请求)

```
personal-task/find-task-list
```

### Response Sample (示例结果)

```
[  
-{  
"workerId": 1,  
"taskId": 1  
},
```

```
-{
  "workerId": 1,
  "taskId": 2
}
```

## personalTaskFindWorkerList

### Description (接口描述)

This interface is used to find a list of worker's ids by task's id which the workers joined in.

Request Method	GET
Request Address	<a href="http://localhost:8080/personal-task/find-worker-list">http://localhost:8080/personal-task/find-worker-list</a>

### Query (参数)

Name	Located in	Required	Description	Example
id	integer	是	The id of the task	2

### Header (参数)

Name	Located in	Required	Description	Example
username	string	是		The token of user

### Responses (返回结果)

Code	Message	Schema
201	succeed	requiement:object

### Request Sample (示例请求)

```
worker/find-by-username
```

### Response Sample (示例结果)

```
[  
-{  
"workerId": 1,  
"taskId": 2  
},  
-{  
"workerId": 2,  
"taskId": 2  
}  
]
```

## personalTaskDelete

### Description (接口描述)

This interface is used to delete one row of personaltask.

Request Method	DELETE
Request Address	<a href="http://localhost:8080/personal-task/delete">http://localhost:8080/personal-task/delete</a>

### Header (参数)

Name	Located in	Required	Description	Example
username	string	是		The token of user

### Responses (返回结果)

Code	Message
201	succeed

### Request Sample (示例请求)

```
personal-task/delete
```

### Response Sample (示例结果)

```
{  
  "code": "201",  
  "message": "succeed"  
}
```

## *changePassword*

### Description (接口描述)

This interface is used to change the password of the user.

Request Method	PUT
Request Address	<a href="http://localhost:8080/changePassword">http://localhost:8080/changePassword</a>

### Responses (返回结果)

code	Message
400	succeed

### Request Sample (示例请求)

```
changePassword
```

### Response Sample (示例结果)

```
{  
  "code": "400",  
  "message": "success"  
}
```

## *seeAllQuestion*

### Description (接口描述)

This interface is used to see all question.

Request Method	GET
----------------	-----

Request Address <http://localhost:8080/question/see-all-question>

### Header (参数)

Name	Located in	Required	Description	Example
username	string	是		The token of user

### Query (参数)

Name	Located in	Required	Description	Example
taskId	integer	是	the id of task	1

### Responses (返回结果)

Code	Message	Schema
201	succeed	question:list<object>

### Request Sample (示例请求)

```
question/see-all-question
```

### Response Sample (示例结果)

```
{
  "code": "200",
  -"Questions":
  [
    -{
      -"question": {
        "id": 3,
        "content": "问卷调查",
        "resource_loading": 0,
        "type": 0,
        "task_id": 6
      },
    }
  ]
}
```

```

- "options":
[ 
-{
"id": 1,
"content": "是",
"questionId": 3,
"openAnswerPermittion": 0,
"optionNumber": 1
},
-{
"id": 2,
"content": "随便写",
"questionId": 3,
"openAnswerPermittion": 1,
"optionNumber": 2
} ],
"resources": null,
"selecteds": null,
"answers": null
}
]
}

```

## *seeHisQuestion*

### Description (接口描述)

This interface is used to see all question.

Request Method	GET
Request Address	<a href="http://localhost:8080/question/see-his-answer">http://localhost:8080/question/see-his-answer</a>

### Header (参数)

Name	Located in	Required	Description	Example
username	string	是		The token of user

### Query (参数)

Name	Located in	Required	Description	Example
------	------------	----------	-------------	---------

Name	Located in	Required	Description	Example
taskId	string	是	the id of the task	6
workerId	string	是	the id of the worker	22

## Responses (返回结果)

Code	Message	Schema
201	succeed	question:list<object>

## Request Sample (示例请求)

```
question/see-his-question
```

## Response Sample (示例结果)

```
{
  "code": "200",
  "Questions": [
    {
      "question": {
        "id": 3,
        "content": "问卷调查",
        "resource_loading": 0,
        "type": 0,
        "task_id": 6
      },
      "options": [
        {
          "id": 1,
          "content": "是",
          "questionId": 3,
          "openAnswerPermittion": 0,
          "optionNumber": 1
        },
        {
          "id": 2,
          "content": "随便写",
          "questionId": 3,
          "openAnswerPermittion": 1,
          "optionNumber": 2
        }
      ],
      "content": "问卷调查"
    }
  ]
}
```

```
"resources": null,
"selecteds": [
{
"id": 1,
"optionId": 1,
"workerId": 22,
"time": "2018-12-04T08:08:37.000+0000"
}
],
"answers": [
{
"id": 1,
"content": "没什么想法",
"questionId": 3,
"workerId": 22,
"time": "2018-12-04T08:08:41.000+0000",
"optionId": 2
}
]
}
]
```