

# 浙江大学

## 本科实验报告

课程名称: B/S 体系软件设计

姓 名: 杨心好

学 院: 计算机科学与技术学院

系: 计算机科学与技术系

专 业: 计算机科学与技术

学 号: 3190104804

指导教师: 胡晓军

2021 年 11 月 7 日

# 浙江大学实验报告

课程名称: B/S 体系软件设计 实验类型: 综合型

实验项目名称: 数据集标注网站

学生姓名: 杨心好 专业: 计算机科学与技术 学号: 3190104804

同组学生姓名: \_\_\_\_\_ 指导老师: 胡晓军

实验地点: 浙江大学玉泉校区 实验日期: 2021 年 11 月 7 日

## 一、项目背景

随着信息化时代的发展，人工智能成为了热门的研究领域，也是将来研究和发展的方向。机器学习作为人工智能中的重要研究领域，近年来也正飞速发展，并成为了非常热门的学科领域。数据集的提供对机器学习的训练有着尤为关键的作用，一个好的数据集标注工具能提高数据集标注的准确性和效率。本项目的目标便是设计一个用于及其学习的图像数据标注网站，实现用户登录后可以进行图片、视频批量上传，视频能够转换为多张图片；任务发布者可以选择图片创建标注任务，其他人可以领取任务进行数据标注；标注结果支持导出多种数据集格式等功能。本项目是 2021-2022 秋冬学期《B/S 体系软件设计》的课程作业，要求开发完整，设计界面友好，提供必要的设计文档和使用手册等文档。

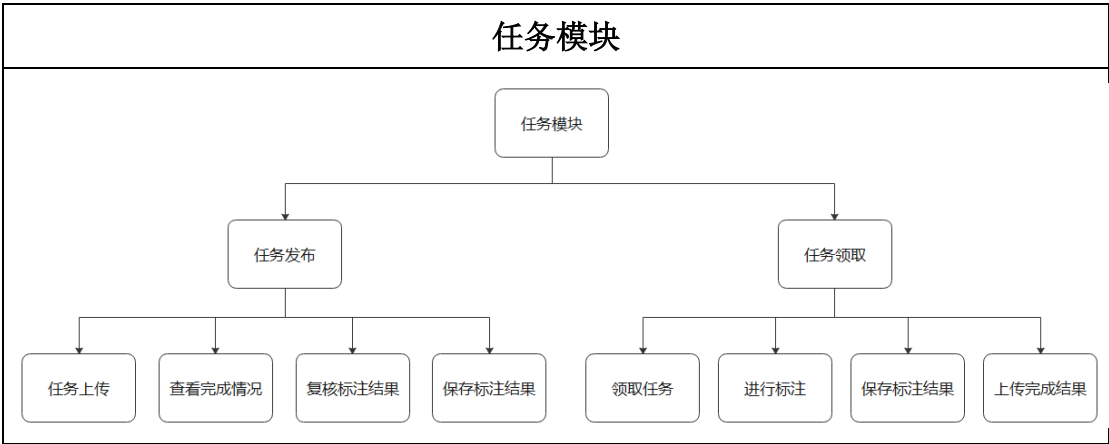
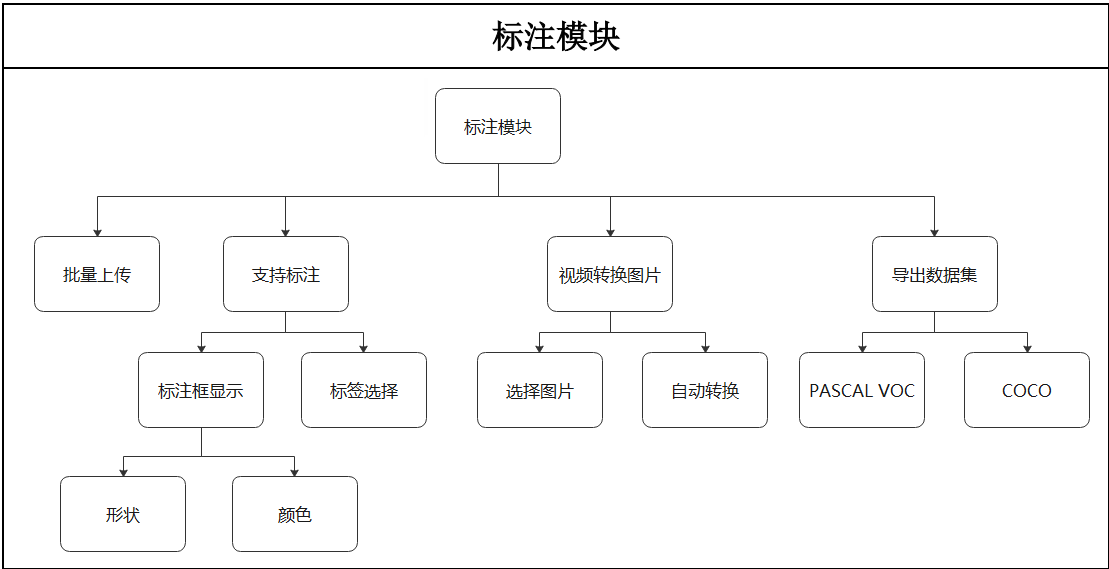
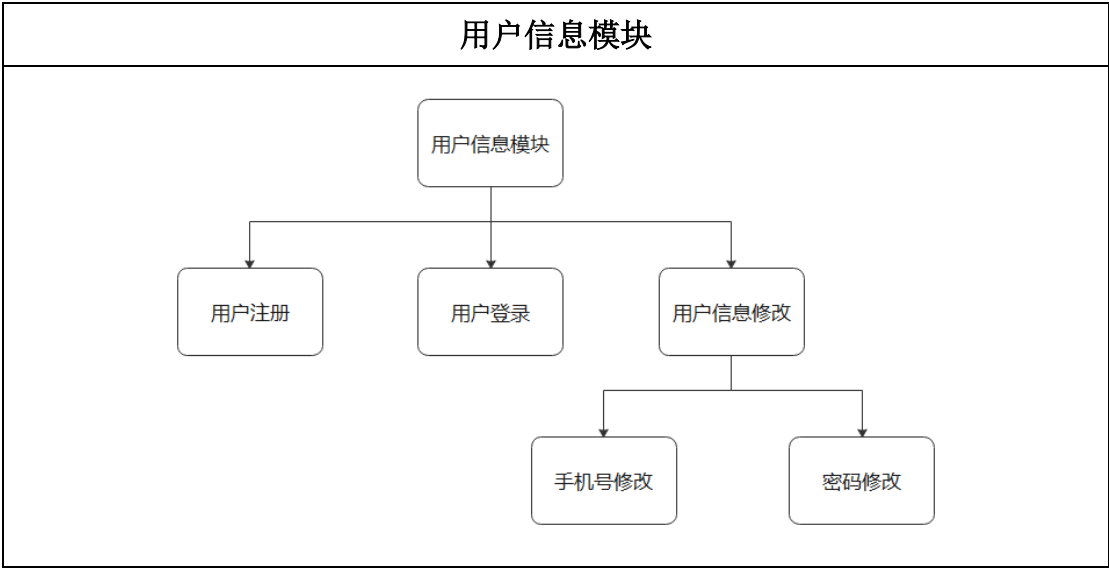
## 二、需求分析

### 2.1 功能性需求

本项目为 B/S 架构的 web 应用，对于用户需要实现如下功能性需求：

- 用户注册（需要用户名，密码，邮箱，手机号等信息，其中用户名和邮箱需要保证唯一性）
- 用户登录
- 用户信息修改
- 批量上传图片、视频
- 可以领取任务进行数据标注，数据标注结果能进行保存
- 作为任务发布者时可以创建标注任务
- 作为任务发布者时可以对标注结果进行复核（增强功能）
- 任务完成情况查看

其中，将用户注册、用户登录、用户信息修改归于用户信息模块，将任务相关的任务发布、任务领取归为任务模块，对于具体的单个标注任务归为标注模块。各个模块的功能需求如下：



## 2.2 非功能性需求

### 2.2.1 性能需求

- 当用户进行登录及其他任何操作时，系统能够及时反应，反应的时间在 1s 以内；
- 每个页面在一般情况下在 1s 内能够加载完毕，高峰期在 6s 内加载完毕；
- 系统在一周内不超过一次维护与重启；
- 目前的各种主流浏览器均能够正常访问本系统；
- 能在不同性能的电脑上经过配置后正常运行；
- 系统应保证运行稳定，避免出现崩溃；
- 系统应该能及时检测出各种非正常情况，如无法连接数据库、输入错误等，并及时反馈给用户，避免用户长时间等待；
- 系统应能保证至少 100 人的并发访问。

### 2.2.2 输入输出需求

- 在用户注册和登录时，应对其输入的数据进行有效性检查，同时应该确保其安全性；
- 标注的标注框应该能够实时显示，同时区分不同类别的物品标注框的颜色；
- 数据库中的数据，如任务完成情况等统计量，需要能够以可视化等方式没关而准确地显示；
- 同时，系统应通过程序控制出错几率，减少系统因用户人为的错误引起的破坏；
- 开发者应该尽量周全地考虑各种可能发生的问题和情况，尽可能降低出错的可能性。

### 2.2.3 数据管理需求

- 系统应具备加密登录、数据加密传输等安全方面的保障，保证数据在不同系统间传输过程中的保密性与安全性；
- 系统应具有数据库维护功能，包括数据导入/导出管理，数据库结构维护、数据恢复功能和性能监测。
- 系统应具有数据库管理功能，包括系统配置与管理、数据存取与更新管理、数据完整性管理和数据安全性管理
- 系统应具有数据库的查询和操纵功能，该功能包括数据库检索和修改。

## 三、系统技术与架构设计

### 3.1 项目技术选型

该项目采用前后端分离的 Web 开发技术，前端技术栈为 Vue.js + Element-UI 组件，后端技术栈为 Django 框架，数据库使用 MySQL，并使用 Postman 作为测试工具。项目技术选型如下所示：

- 前端：Vue + Vuex + Element UI 组件库 + axios + Vue router
- 后端：Django 框架 + Django REST Framework + Django Cor Headers
- 数据库：MySQL
- 测试工具：Postman 用于测试后端接口数据正确性
- 服务器：MQTT

### 3.2 项目主要技术介绍

#### 3.2.1 前端技术框架

项目中主要使用了 Vue.js + axios + Element-UI+ Vue router 组件库作为前端开发的技术栈。

Vue 是一套用于构建用户界面的渐进式框架。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，

还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，Vue 也完全能够为复杂的单页应用提供驱动。

element ui 是一款设计和交互非常优秀的前端 UI 框架。基于 Vue 2.0，同时也有基于 React 和 Angular 的实现。

Axios 是一个基于 promise 网络请求库，作用于 node.js 和浏览器中，它是 isomorphic 的(即同一套代码可以运行在浏览器和 node.js 中)。在服务端它使用原生 node.js http 模块，而在客户端 (浏览端) 则使用 XMLHttpRequests。

Vue Router 是 Vue.js (opens new window)官方的路由管理器。它和 Vue.js 的核心深度集成，让构建单页面应用变得易如反掌。

### 3.2.2 后端技术框架

Django 是高水准的 Python 编程语言驱动的一个开源模型、视图、控制器风格的 Web 应用程序框架，它起源于开源社区。使用这种架构，程序员可以方便、快捷地创建高品质、易维护、数据库驱动的应用程序。这也正是 OpenStack 的 Horizon 组件采用这种架构进行设计的主要原因。另外，在 Django 框架中，还包含许多功能强大的第三方插件，使得 Django 具有较强的可扩展性。

Django 的主要目的是简便、快速的开发数据库驱动的网站。它强调代码复用，多个组件可以很方便的以“插件”形式服务于整个框架，Django 有许多功能强大的第三方插件，甚至可以很方便的开发出自己的工具包。这使得 Django 具有很强的可扩展性，同时也强调快速开发和 DRY(Do Not Repeat Yourself)原则。

### 3.2.3 MQTT 服务器

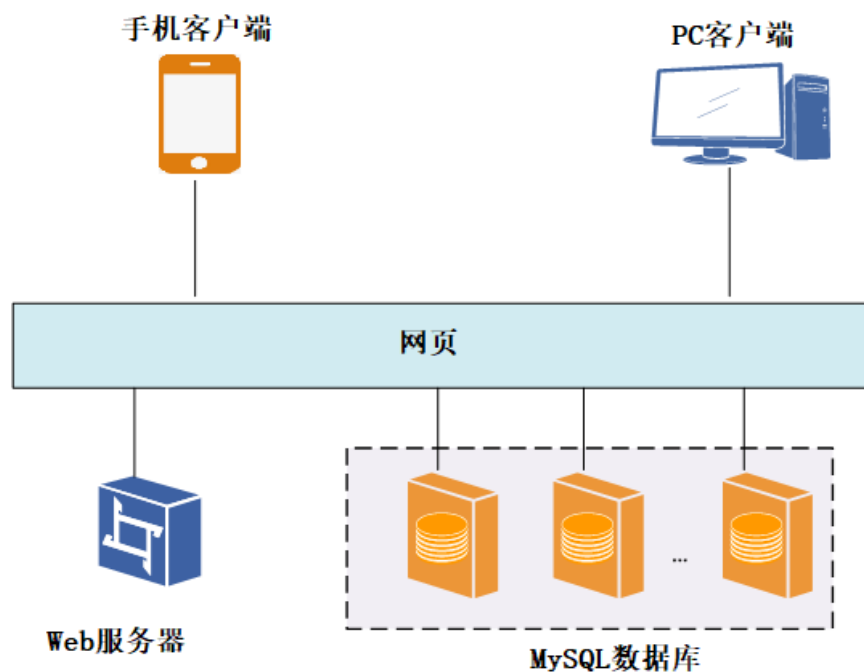
MQTT (Message Queuing Telemetry Transport, 消息队列遥测传输协议)，是一种基于发布/订阅 (publish/subscribe) 模式的"轻量级"通讯协议，该协议构建于 TCP/IP 协议上，由 IBM 在 1999 年发布。MQTT 最大优点在于，可以以极少的代码和有限的带宽，为连接远程设备提供实时可靠的消息服务。作为一种低开销、低带宽占用的即时通讯协议，使其在物联网、小型设备、移动应用等方面有较广泛的应用。

### 3.3 项目架构图

该项目为前后端分离的 web 开发项目，系统的总体架构图及前后端架构图分别如下。

#### 3.3.1 系统总体架构图

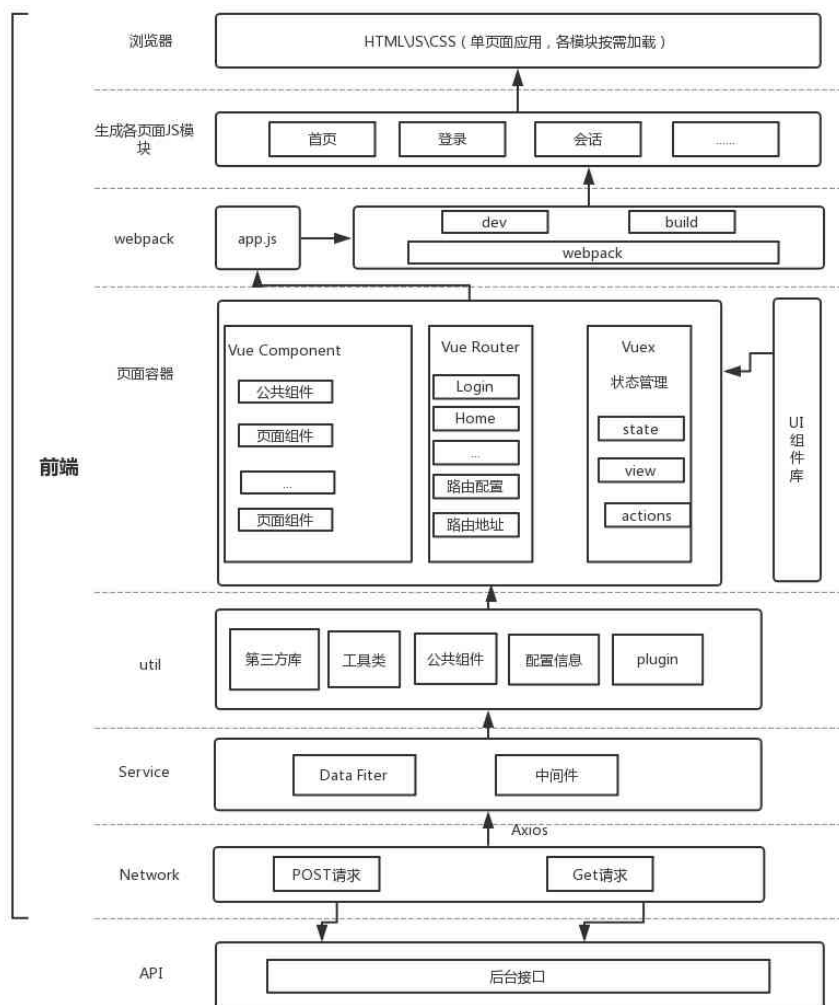
系统的总体框架如下图所示，其中客户端支持 PC 和手机端等多种终端访问，服务端主要有后端服务器与数据库进行交互，后端服务器可以接受网页发出的 HTTP 请求并处理，并及时存储在数据库中。



#### 3.3.2 前端架构图

该项目的前端主要采用基于 Node.js 的 Vue 框架，并采用开元项目 Element U 作为 UI 组件框架，用 Vue Router 来管理页面的路由，前端总体架构可以用下图表示：





### 3.3.3 后端架构图

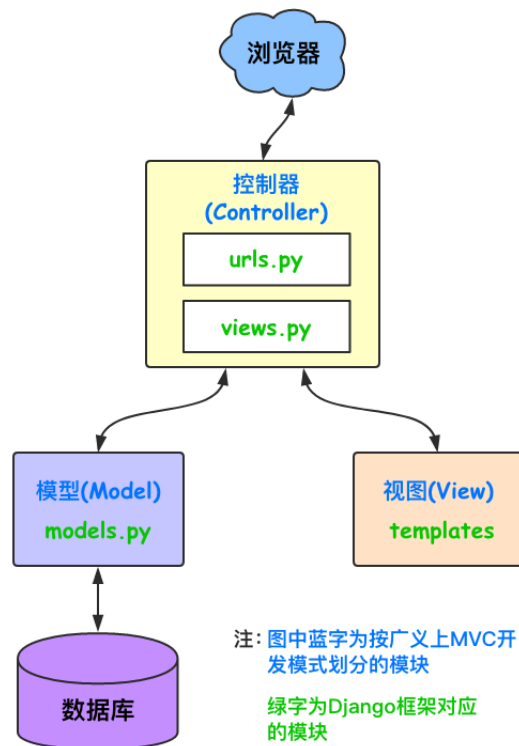
该项目后端主要使用 Python 的 Django 框架编写而成。

Django 框架的设计模式借鉴了 MVC 框架的思想,分成三部分:模型(Model)、视图(View)和控制器(Controller), 来降低各个部分之间的耦合性。

Django 框架的不同之处在于它拆分的三部分为: Model (模型)、Template (模板) 和 View (视图), 也就是 MTV 框架。

- Model(模型): 负责业务对象与数据库的对象(ORM)
- Template(模版): 负责如何把页面展示给用户
- View(视图): 负责业务逻辑, 并在适当的时候调用 Model 和 Template

此外, Django 还有一个 urls 分发器, 它的作用是将一个个 URL 的页面请求分发给不同的 view 处理, view 再调用相应的 Model 和 Template。后端 Django 的框架可以用下图表示:



### 3.4 运行环境

项目对运行环境的部署要求及版本如下：

- Python 3.8
- Node.js v14
- MySQL 8.0
- Django 3.x
- Vue 2.0 及以上

## 四、数据库设计与 ER 图

### 4.1 数据表设计

本项目数据库共设计了 6 个表，分别存储用户信息、任务信息、图片信息、标注文件信息、标注框信息、标注标签信息，具体表结构设计如下：

#### 4.1.1 user 表

- User 表用于保存用户信息

字段名	类型	描述	备注
id	int	用户的 id	主键，自增
username	varchar(32)	用户名	UNIQUE,MIN(6)
password	varchar(32)	用户密码	MIN(6)
email	varchar(32)	用户邮箱	UNIQUE
phone	varchar(16)	用户手机号	
gender	char	用户性别	
birth	varchar(8)	用户生日	

#### 4.1.2 task 表

- task 表用于保存任务信息

字段名	类型	描述	备注
id	int	任务 id	主键，自增
name	varchar(32)	任务名	
description	varchar(256)	任务描述	
publisher_id	int	发布任务用户 id	外键
accepter_id	int	接受任务用户 id	外键
picfile_name	varchar(128)	图片文件夹名	
lablefile_name	varchar(128)	标注文件夹名	
picfile_path	varchar(128)	图片文件夹路径	
lablefile_path	varchar(128)	标注文件夹路径	
finish_num	int	已完成数量	
status	int	是否被领取	
start_date	time	开始时间	
end_date	time	完成时间	

### 4.1.3 image 表

- image 表用于保存图片信息

字段名	类型	描述	备注
id	int	图片 id	主键，自增
img_name	varchar(128)	图片名	
width	int	图片宽度	
height	int	图片高度	
file_name	varchar(128)	文件夹名	
annotated	int	是否已被标注	
anno_num	int	标注框数量	
file_path	varchar(128)	图片文件夹路径	
anno_id	int	对应的标注文件	外键，SET_NULL

### 4.1.4 Annotation 表

- Annotation 表用于保存标注文件信息

字段名	类型	描述	备注
id	int	标注文件 id	主键，自增
anno_name	varchar(32)	标注文件名	
img_id	int	对应的图片 id	外键，CASCADE
format	varchar(32)	数据集格式	
file_path	varhchar(128)	标注文件夹路径	

### 4.1.5 Object 表

- Object 表用于保存标注目标信息

字段名	类型	描述	备注
id	int	标注框 id	主键，自增
anno_id	int	所在标注文件	外键，CASCADE
label_id	varchar(32)	目标标签 id	外键

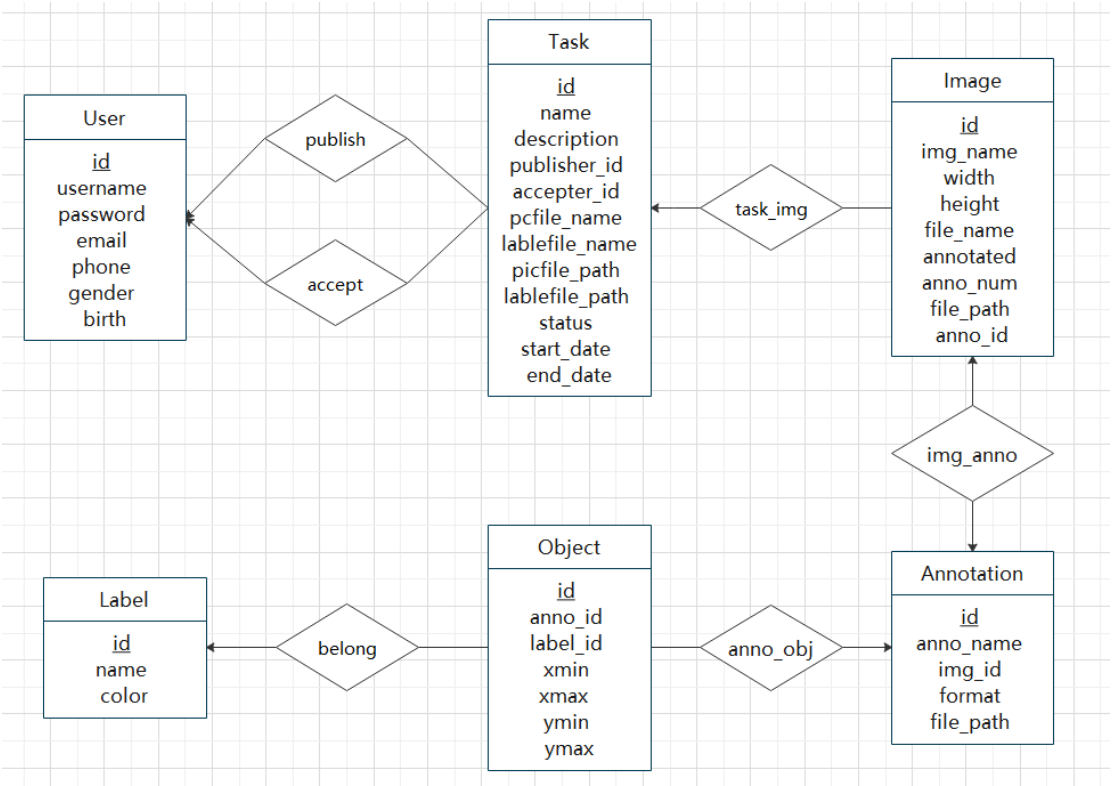
xmin	int	目标左侧 x 坐标	
xmax	int	目标右侧 x 坐标	
ymin	int	目标下边 y 坐标	
ymax	int	目标上边 y 坐标	

#### 4.1.6 Label 表

- Label 表用于保存标注标签信息

字段名	类型	描述	备注
id	int	标注标签 id	主键，自增
name	varchar(32)	标注标签名称	
color	varchar(32)	标注框对应颜色	

#### 4.2 ER 图



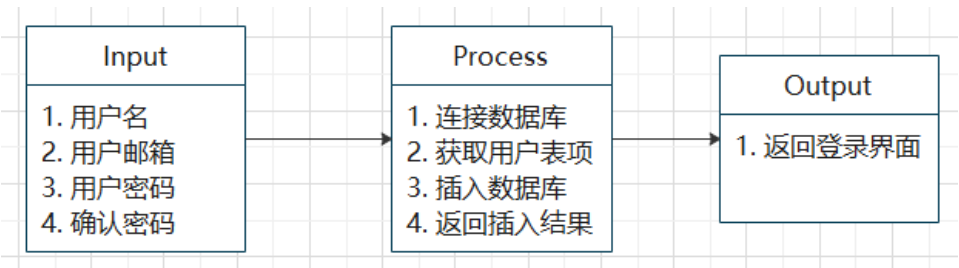
## 五、基本执行流程

### 5.1 用户注册

#### 5.1.1 模块概述

用户注册模块中，用户需要输入注册的用户名和邮箱，并输入密码，同时需要再次输入密码进行确认。其中，用户名和邮箱不能与现存用户重复，用户名和密码至少应有 6 个字符，密码与确认密码应该相同。

#### 5.1.2 IOP 图



#### 5.1.3 输入输出项

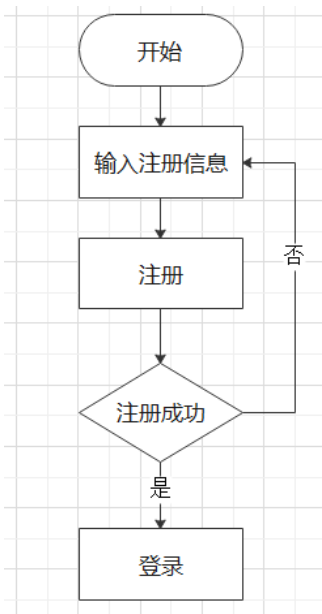
##### ● 输入

名称	标识	类型及格式	输入方式
用户名	user_name	string	外部输入
用户邮箱	email	string	外部输入
用户密码	password	string	外部输入
确认密码	confirm	string	外部输入

##### ● 输出

名称	标识	类型及格式	输出方式
注册结果	register_result	bool	由模块输出

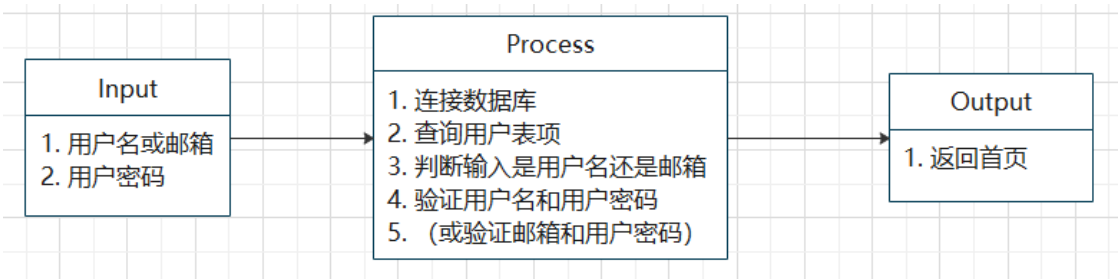
5.1.4 流程图



5.2 用户登录

5.2.1 模块概述

该模块为用户提供登录功能，用户可以选择用户名+密码或邮箱+密码两种登录方式。



5.2.2 输入输出项

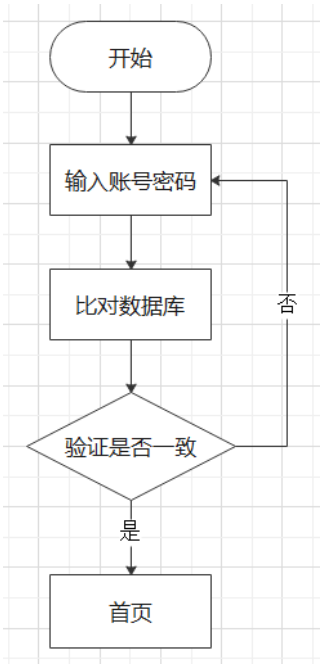
● 输入

名称	标识	类型及格式	输入方式
用户名	name	string	外部输入
邮箱	email	string	外部输入
密码	password	string	外部输入

● 输出

名称	标识	类型及格式	输出方式
登录结果	login_result	bool	模块输出

5.2.3 流程图

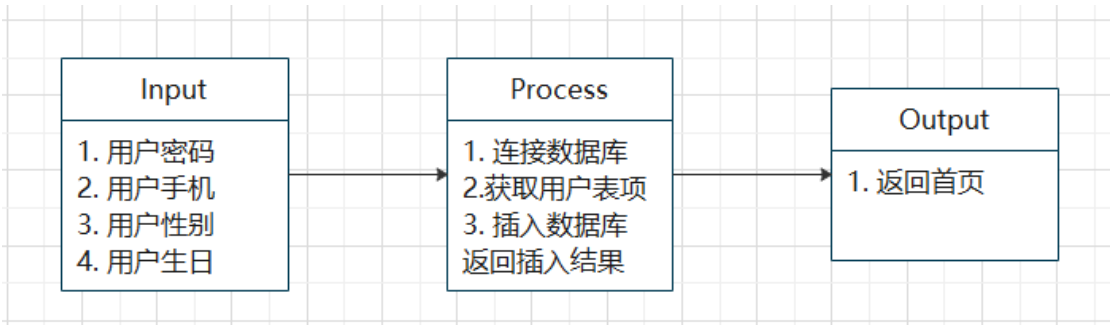


5.3 用户修改信息

5.3.1 模块概述

该模块提供用户修改的功能，用户可以对密码进行修改，同时完善手机号、性别、生日等其他注册时没有要求填写的信息。

5.3.2 IPO图





5.3.3 输入输出项

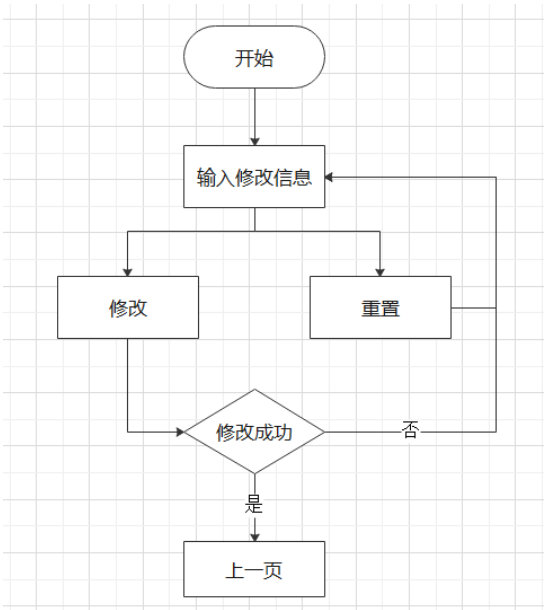
● 输入

名称	标识	类型及格式	输入方式
用户密码	password	string	外部输入
用户手机	phone	string	外部输入
用户性别	gender	string	外部输入
用户生日	birth	calendar	外部输入

● 输出

名称	标识	类型及格式	输出方式
修改结果	modify_result	bool	模块输出

5.3.4 流程图

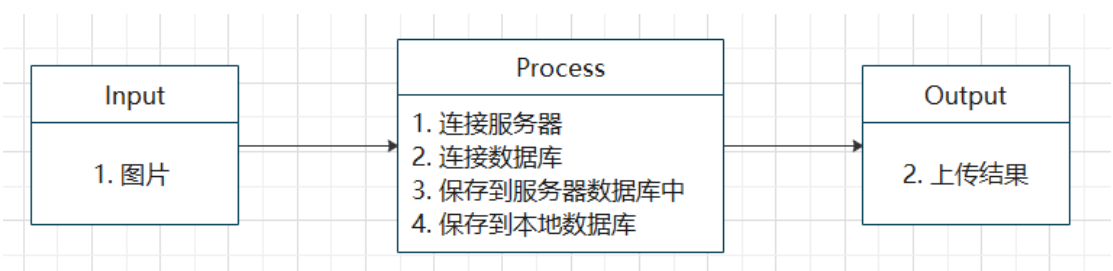


5.4 上传图片/视频

5.4.1 模块概述

该模块提供（批量）上传图片及视频功能，并支持将视频按照一定规则转换为图片，图片能够保存在服务器。

5.4.2 IPO图



5.4.3 输入输出项

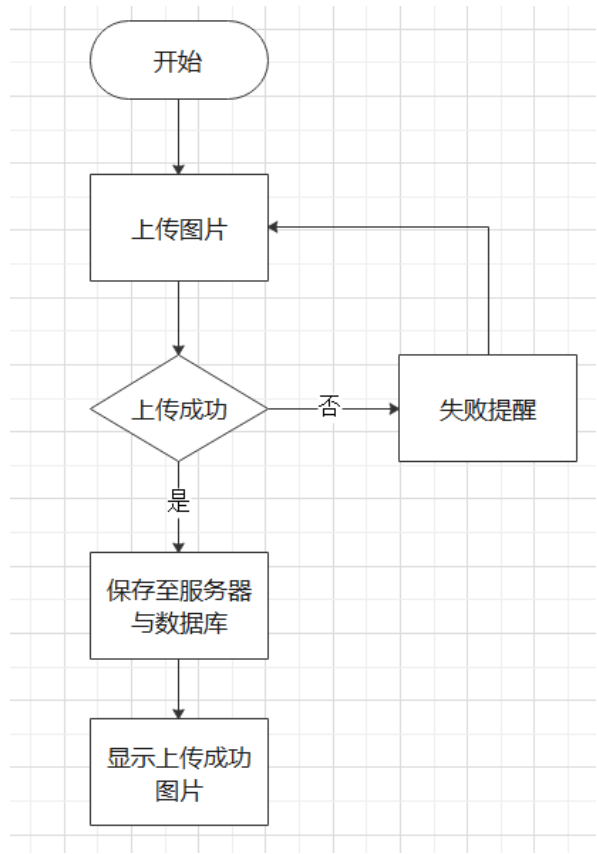
● 输入

名称	标识	类型及格式	输入方式
图片	img_id	int	外部输入
文件夹	file_path	string	外部输入

● 输出

名称	标识	类型及格式	输出方式
上传结果	upload_result	bool	模块输出

5.4.4 流程图

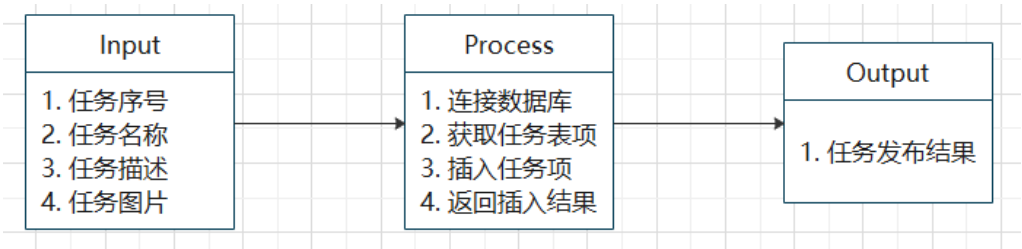


5.5 任务发布

5.5.1 模块概述

任务发布模块为用户提供任务发布的功能，需与数据库进行交互。

5.5.2 IPO 图



5.5.3 输入输出项

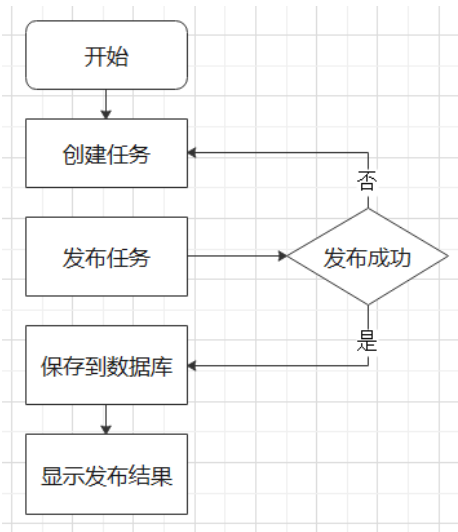
● 输入

名称	标识	类型及格式	输入方式
任务序号	task_id	int	外部输入
任务名称	task_name	string	外部输入
任务描述	description	string	外部输入
任务图片	img_id	list	外部输入

● 输出

名称	标识	类型及格式	输出方式
发布结果	publish_result	bool	模块输出

5.5.4 流程图

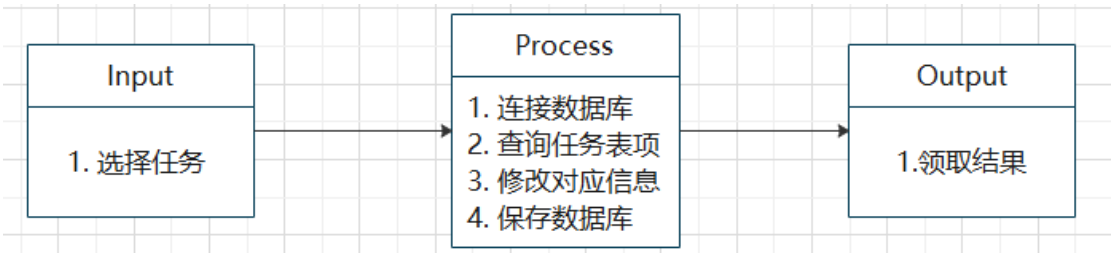


5.6 任务领取

5.6.1 模块概述

该模块为用户提供任务领取功能，用户可以领取任务进行标注，标注完成后能够导出标注结果并提交。

5.6.2 IPO 图



5.6.3 输入输出项

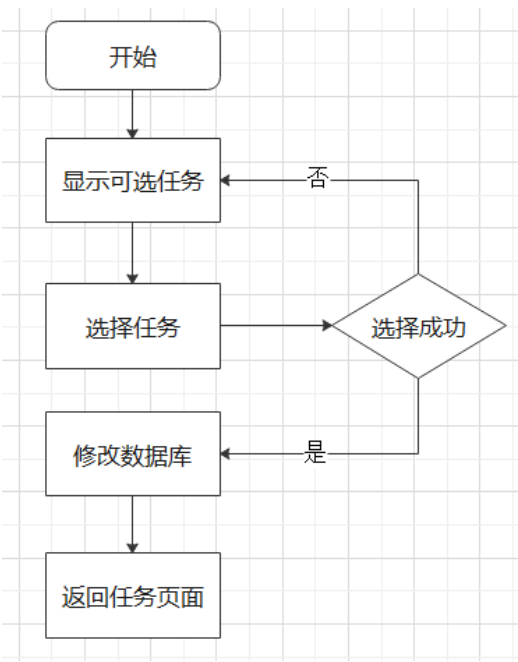
● 输入

名称	标识	类型及格式	输入方式
任务序号	task_id	int	外部选择

● 输出

名称	标识	类型及格式	输出方式
领取结果	receive_result	bool	模块输出

5.6.4 流程图

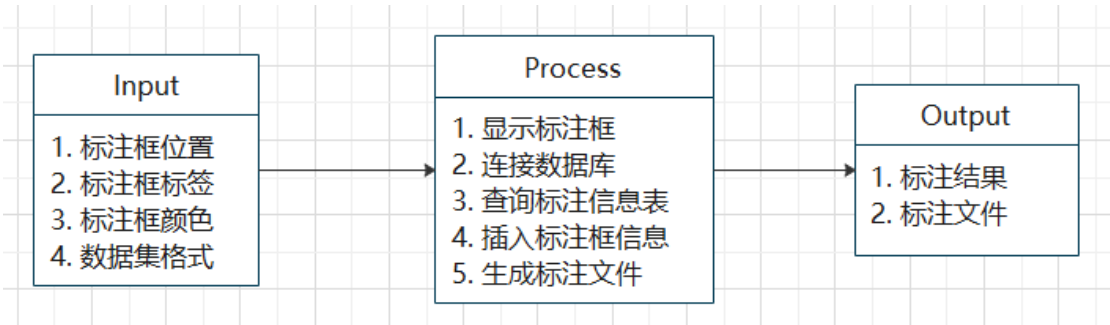


## 5.7 进行标注

### 5.7.1 模块概述

该模块为用户提供标注功能，用户能够在图片上进行标注，选择标签，并选择数据集格式，将标注结果导出。标注完成的图片能够显示。

### 5.7.2 IPO 图



### 5.7.3 输入输出项

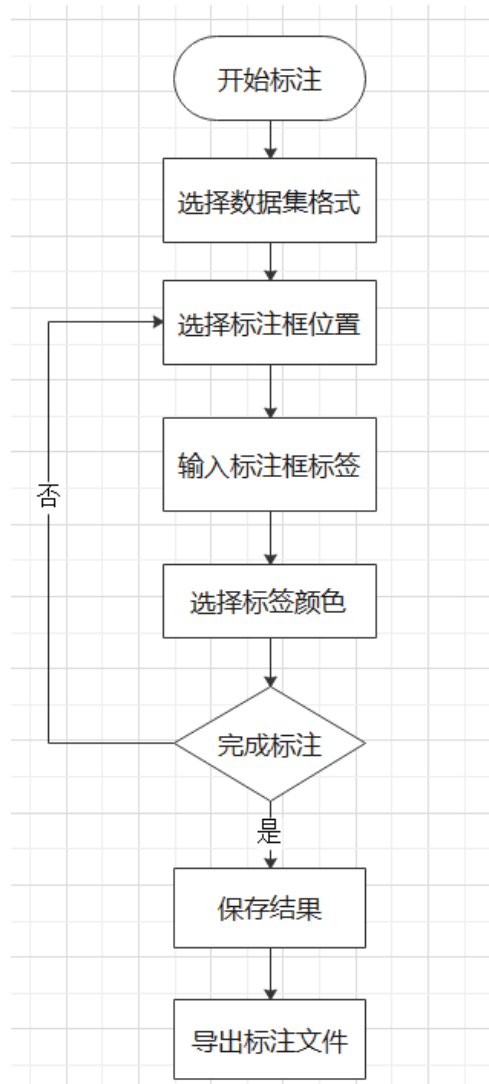
#### ● 输入

名称	标识	类型及格式	输入方式
左边 x 坐标	xmin	int	外部选择
右边 x 坐标	xmax	int	外部选择
上面 y 坐标	ymin	int	外部选择
下面 y 坐标	ymax	int	外部选择
标签名	label_name	string	外部输入
标签颜色	label_color	string	外部选择
结果保存路径	filepath	string	外部输入
数据集格式	format	string	外部选择

#### ● 输出

名称	标识	类型及格式	输出方式
标注结果	anno_result	bool	模块输出
标注文件	anno_id	int	模块输出

#### 5.7.4 流程图

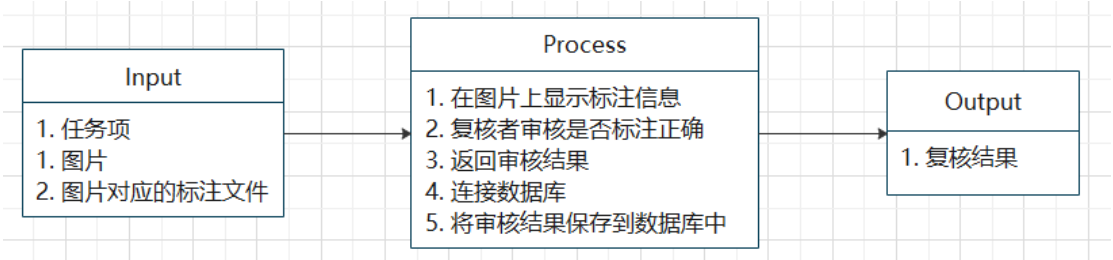


### 5.8 结果复核

#### 5.8.1 模块概述

该模块为任务发布者提供复核功能，可以复核标注结果是否正确，如果不正确，可以退回或自己重新标注。

5.8.2 IPO 图



5.8.3 输入输出项

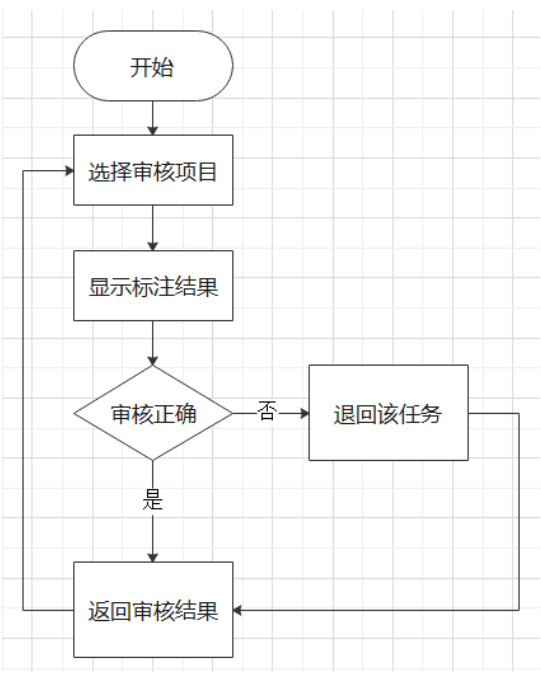
● 输入

名称	标识	类型及格式	输入方式
任务序号	task_id	int	外部选择
图片序号	img_id	int	外部选择
标注文件	anno_id	int	与图片相联

● 输出

名称	标识	类型及格式	输出方式
审核结果	judge_result	bool	模块输出
审核描述	judge_description	string	模块输出

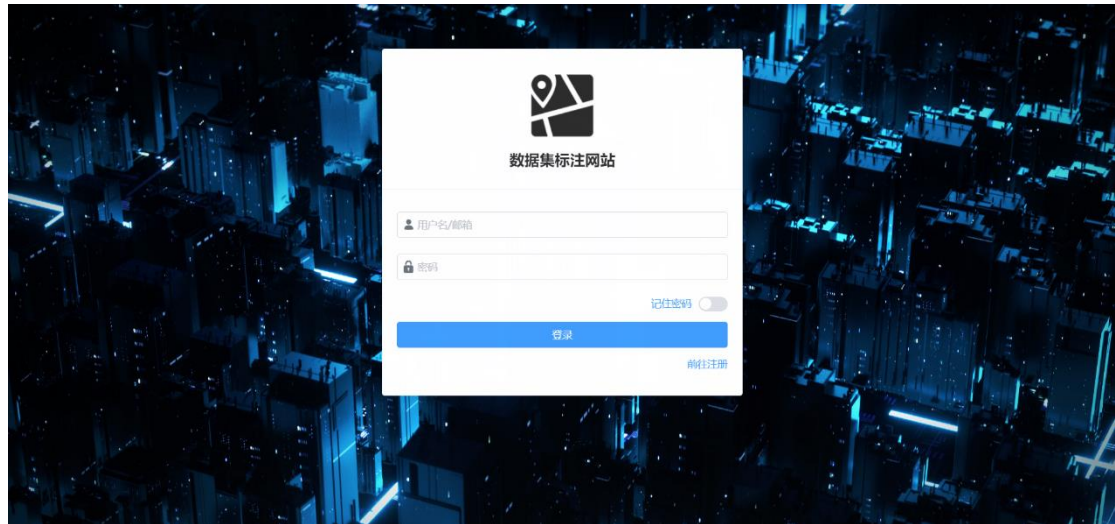
5.8.4 流程图



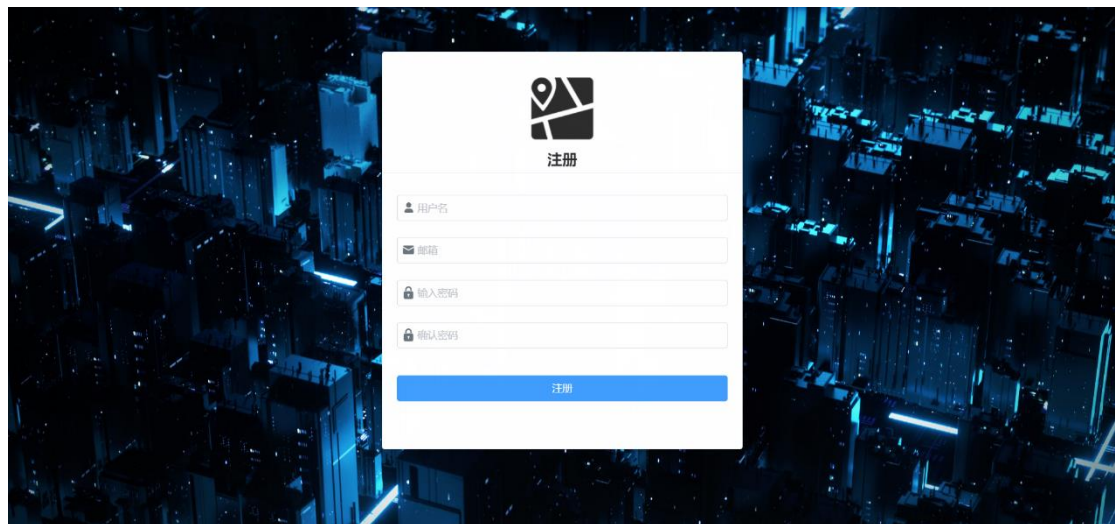
## 六、初步界面设计

本阶段初步实现了几个前端界面，主要用登录、注册、任务发布、任务领取、首页等界面。由于项目仍处于开发阶段，最终成品和界面原型可能存在较大差异，以最终成品为主。

### ● 登录界面

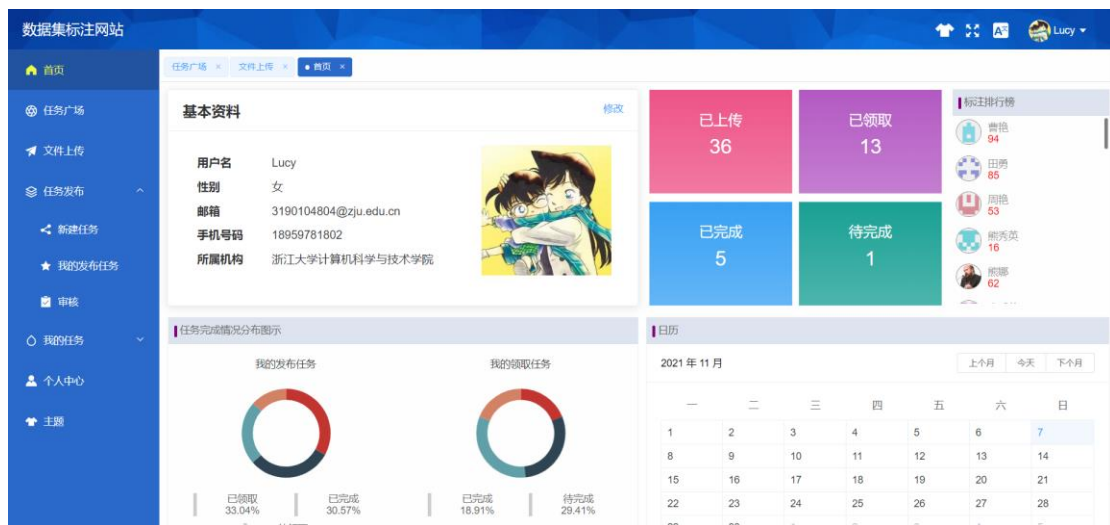


### ● 注册界面





## ● 网站首页



## ● 修改信息界面

数据标注网站

任务广场 个人中心

个人中心

个人资料

\* 用户名: Lucy

\* 邮箱:

\* 手机号:

\* 性别: ☐ 男 ☒ 女

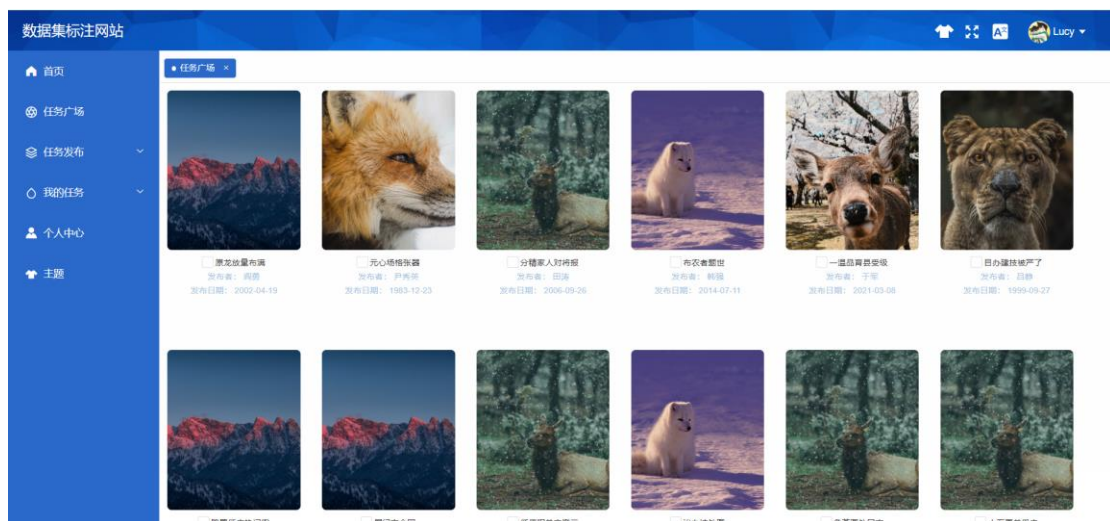
出生日期: 选择日期

保存 重置

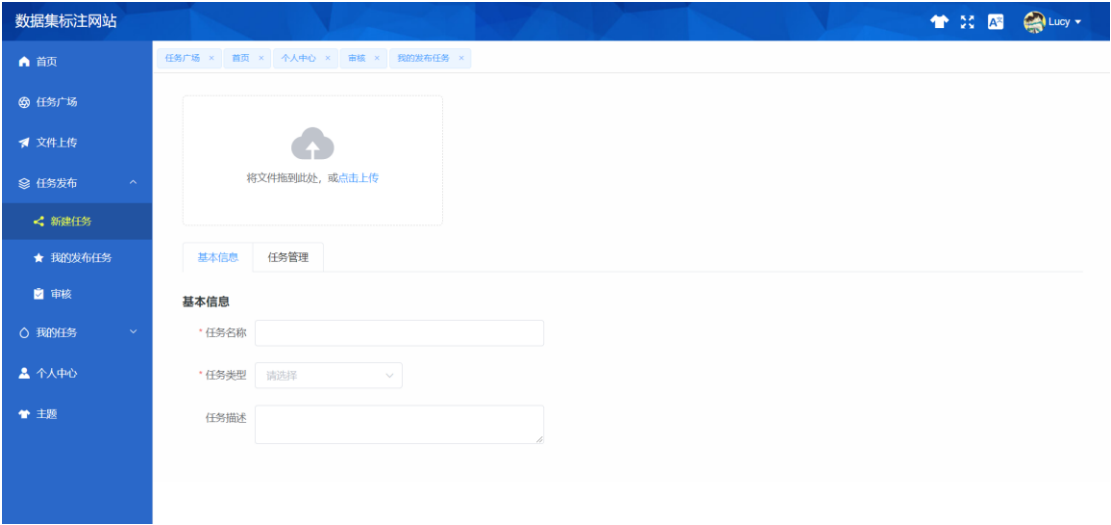
头像

修改

## ● 任务领取界面



● 任务发布界面



七、项目进度安排

时间段	计划进度
2021.10	完成系统的设计与初步开发
2021.11-2021.12 中旬	基本完成系统的前后端代码开发
2021.12 下旬	完成系统的集成与测试，编写各个文档

八、备注

该设计报告完成于中期，实际完成情况与设计报告存在较大差异，以最终成品为准。