北理失物招领平台

**软件需求规格说明书**

**4-6班 第一组**

2019年06月15日 **目 录**

[1. 引言 3](#_Toc142378827)

[1.1 目的 3](#_Toc142378828)

[1.2 适用范围 3](#_Toc142378829)

[1.3 参考资料 3](#_Toc142378830)

[1.4 阅读建议 3](#_Toc142378830)

[2. 系统概述 3](#_Toc142378832)

[2.1 产品描述 3](#_Toc142378833)

[2.2 产品功能 4](#_Toc142378834)

[2.3 一般约束 4](#_Toc142378835)

[3. 功能性需求分类 5](#_Toc142378836)

[3.1 功能描述1-用户层 7](#_Toc142378837)

[3.2 功能描述2-管理员层 1](#_Toc142378838)0

[4. 产品的非功能性需求 1](#_Toc142378839)1

[4.1 外部接口说明 11](#_Toc142378840)

[4.1.1 用户接口 1](#_Toc142378841)1

[4.1.2 软件接口 1](#_Toc142378842)1

[4.2 内部接口说明 1](#_Toc142378843)2

[4.2.1 用户数据接口 1](#_Toc142378844)2

[4.2.2 物品数据接口 1](#_Toc142378844)3

[4.3 属性 1](#_Toc142378845)4

[4.3.1 友好性 1](#_Toc142378846)4

[4.3.2 可维护性 1](#_Toc142378848)4

[4.3.3 可转移/换性 1](#_Toc142378849)4

[5. 数据字典 1](#_Toc142378839)5

[5.1 用户信息 1](#_Toc142378852)5

[5.2 物品信息 1](#_Toc142378852)5

# 引言

## 目的

应甲方（4-6班第三组）提出的需求，为改进目前北京理工大学校园内失物存储柜缺乏管理、使用不便的情况，方便同学们找回自己的失物、归还捡到的物品，建立此平台。

## 适用范围

面向用户：北京理工大学在校生

适用场景：用户查找他人发布的失物/招领信息，发布自己的失物/招领信息

项目形式：网站

适用平台：电脑、手机等能浏览网页的设备

## 参考资料

|  |  |
| --- | --- |
| **资料名称** | **作 者** |
| Django2.2 官方说明文档 | Django Software Foundation |
| 阿里云短信服务、邮件服务说明文档 | 阿里云 |
| 《Django2 待办清单网站》《Django2 登陆注册》 | @\_\_捷佳\_\_ |
| Bootstrap 官方说明文档 | [@mdo](https://twitter.com/mdo" \t "https://v4.bootcss.com/_blank), [@fat](https://twitter.com/fat" \t "https://v4.bootcss.com/_blank) |

## **阅读建议**

产品用户建议阅读2.2、3、4.3节，并参阅同目录下的视频演示。

开发者、维护者建议阅读2.1、3、4.1、4.2节。

# 系统概述

## 产品描述

**开发背景**：目前，北理校内仅有的失物招领方式是存在于各个建筑中少量的失物柜，没有安全保障且十分分散，丢失物品、捡到物品的同学没有方便的途径认领、归还物品。因此，建立一个能让所有同学分享信息的交流平台是十分必要的。

**系统概况**：平台为网页形式，主页上分类列出他人发布的失物/招领信息。所有人浏览、检索，注册用户可以发布信息、响应他人的信息。

**开发环境：**

|  |  |
| --- | --- |
| 系统 | Windows 10 |
| 语言 | Python 3.7， Html5，CSS，JavaScript |
| 框架 | Django 2.2 |
| 数据库 | SQLite |

**开发工具：**

|  |  |
| --- | --- |
| 主编辑器 | Pycharm 2019，Atom |
| 命令行工具 | Cmd，Anaconda Prompt |
| 调试工具 | Chrome |
| 协作工具 | Github Desktop |
| 统计分析工具 | Stastic |

## 产品功能

|  |  |  |
| --- | --- | --- |
| 编号 | 模块名称 | 功能描述 |
| 1 | 浏览功能 | 在网站上查看他人发布的信息 |
| 2 | 搜索功能 | 模糊搜索平台上的失物信息 |
| 3 | 登录注册模块 | 平台基本操作，用于用户的登录与注册 |
| 4 | 个人主页面 | 管理个人信息，可进行对信息的修改 |
| 5 | 失物发布功能 | 发布丢失物品的相关信息 |
| 6 | 拾取物品发布功能 | 发布拾取的物品的相关信息 |

## 注解：

## （1）功能1、2无需登录，所有浏览者均可使用。

## （2）功能1（浏览）包含了按物品分类进行检索。

## （3）功能2（搜索）为整合性搜索，用户可以根据物品的任一项属性（包括名称、类型、发布者、描述、发布时间、地点）搜索到物品。

## （4）功能4（个人主页面）的信息修改包含对一般信息的修改（用户名、邮箱、学号、手机号等），可直接修改；对密码的修改，需验证旧密码，且修改成功后须重新登录。

## 一般约束

* + 1. 硬件约束

开发使用的设备为两个程序员的个人笔记本电脑，且均为Windows10平台，为多环境的测试带来不便。

* + 1. 时间约束

除去开始的需求获取、分析、修正的时间，总的开发时间仅有不到三周。

* + 1. 人员约束

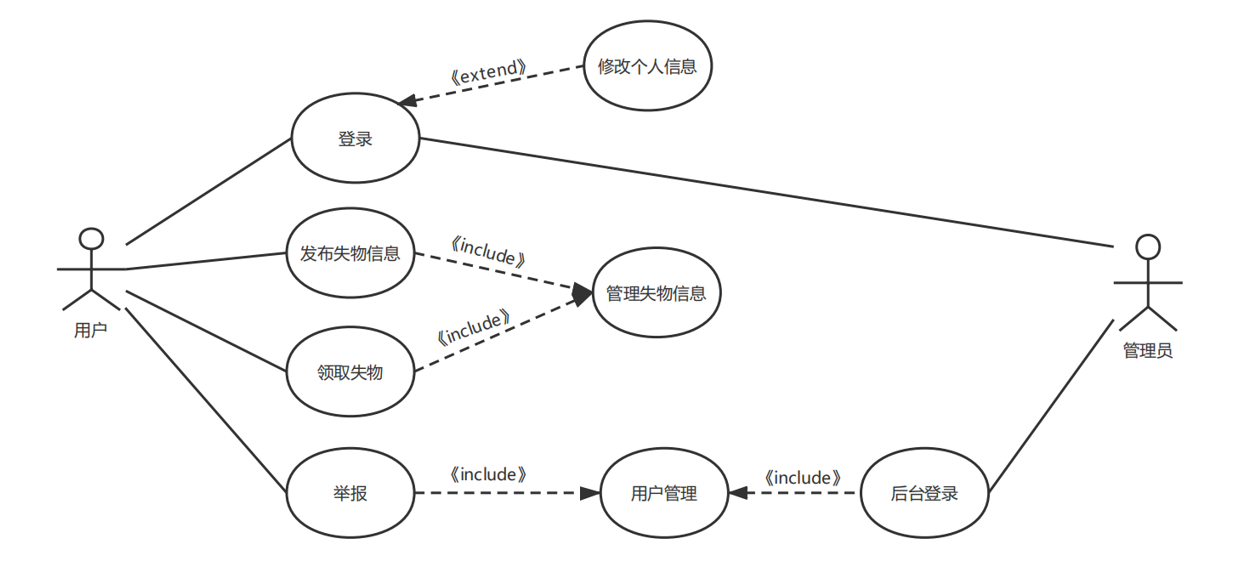
小组中仅有两人有开发经验，工作量很大。

* + 1. 其他约束

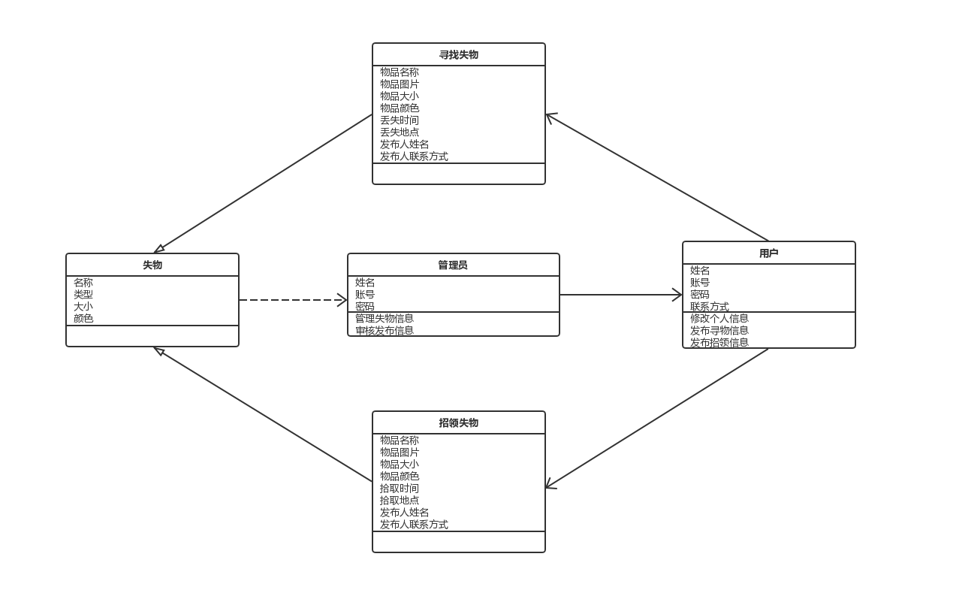
开发同期小组成员有别的科目学习、考试任务。

# 功能性需求分类

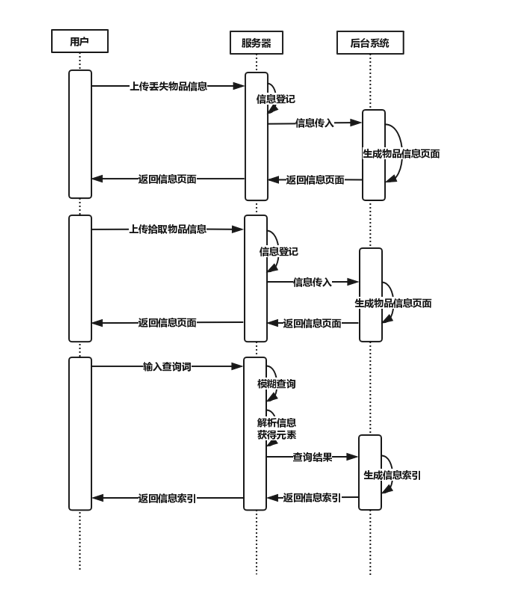
用例图：



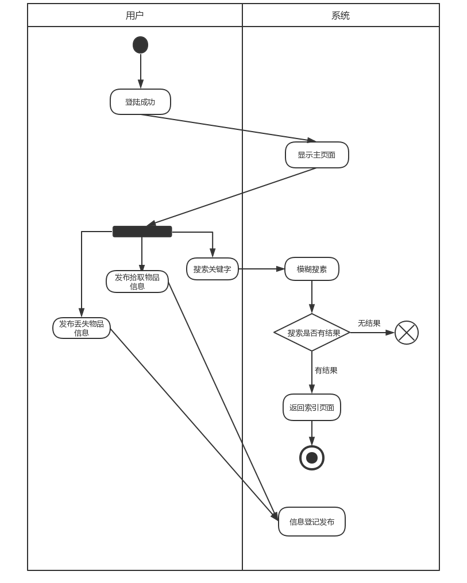
类图：



动态UML图-交互图：



动态UML图-活动图：



## 功能描述1-用户层

前注：物品分为失物、招领两大板块，各自拥有浏览、分类检索、页内搜索功能，且板块内功能仅对应所在版块（如在失物板块进行搜索，结果仅包含失物），此后描述中不再说明其对应的板块。

### 浏览

顶端导航栏右侧可以切换板块；网页主体部分左侧为分类侧边栏，可进行分类检索；右侧为按要求显示出的信息。



### **搜索**

通过网页顶端导航栏左侧的搜索栏可以进行页内搜索，可以通过物品的各项属性（包括名称、类型、发布者、描述、发布时间、地点）进行搜索。



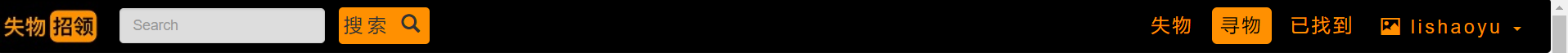
### **注册登录**

通过（1）网页顶端导航栏右侧的注册按钮（2）登录页面的注册按钮可进入注册页面，按照提示输入符合要求的信息可以注册为网站用户。

通过（1）网页顶端导航栏右侧的登录按钮（2）点击需要登录才能使用的功能（如发布）可进入登录页面，输入正确的用户名和密码即可登录。







### **发布信息**

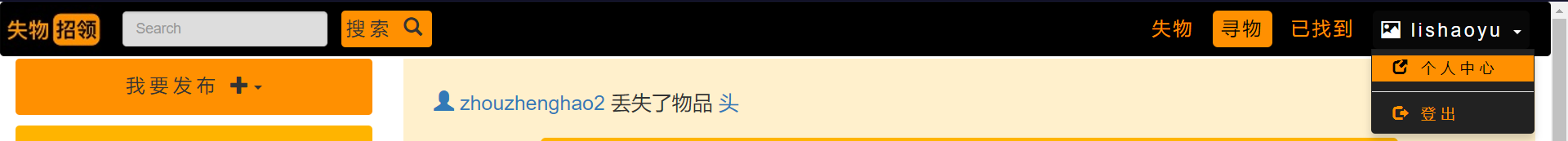
登录后通过左侧侧边栏的按钮进入发布页面，输入符合要求的信息后可以发布信息。





### 个人信息管理

登录后可通过网页顶端导航栏右侧的个人中心按钮进入个人中心，在其中查看信息。通过点击个人中心里的编辑信息和修改密码可以进入对应页面修改信息。修改密码之后需要重新登录。



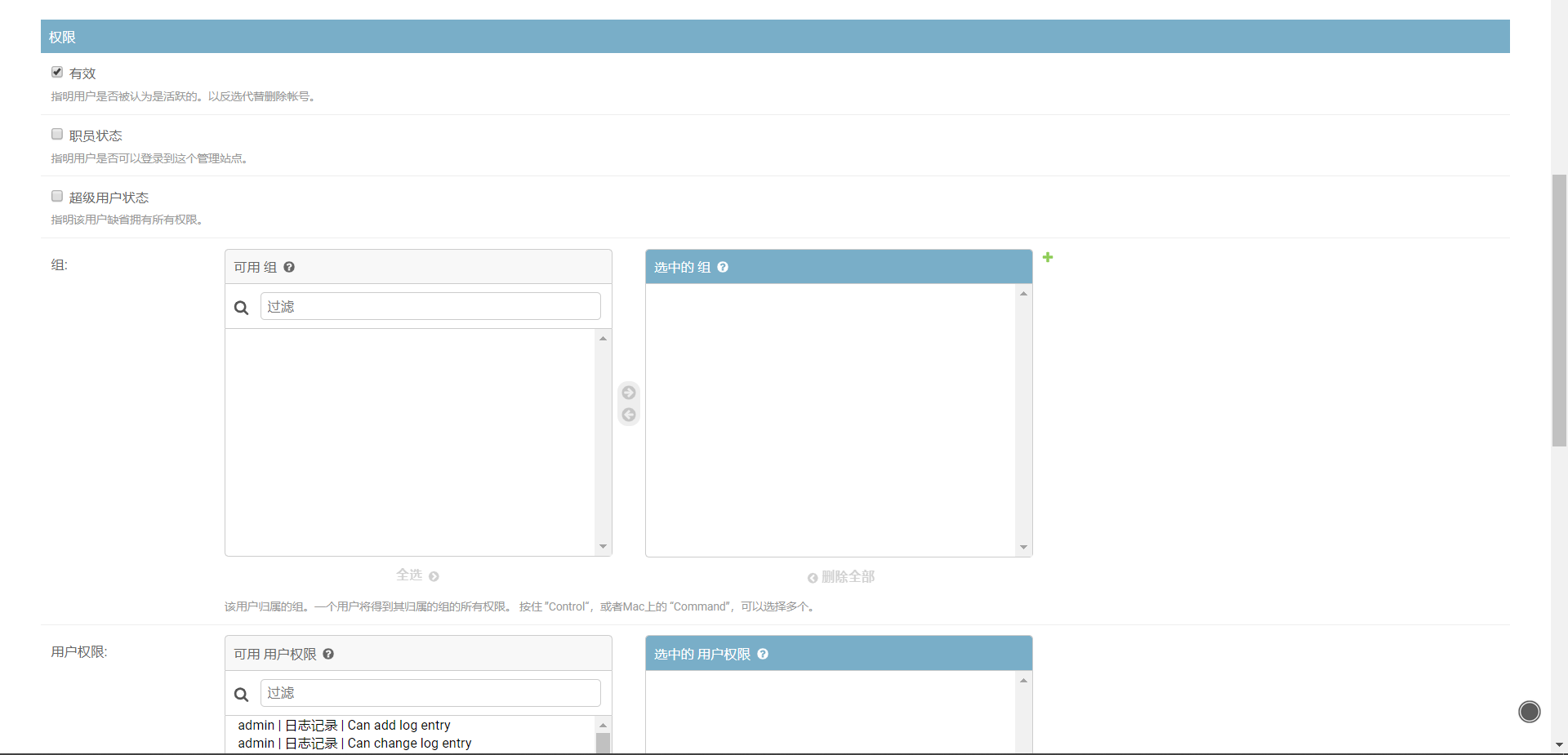
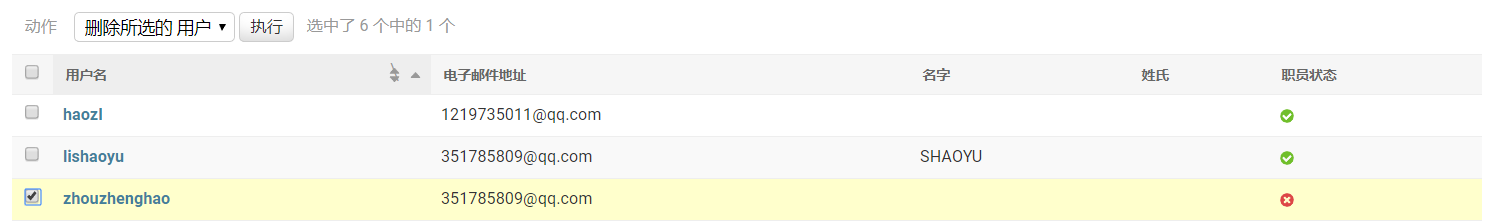
## 功能描述2-管理员层

### **基础功能**

管理员（超级用户）可使用普通用户可使用的全部功能。

### **用户管理功能**

管理员在后台界面可以对用户进行修改、删除、权限赋予操作。



# 产品的非功能性需求

## 外部接口说明

前注：因尚未付费租用相关服务的服务器/流量，以下部分服务仅完成了接口的编写、审核，暂时不能使用。

## 用户接口

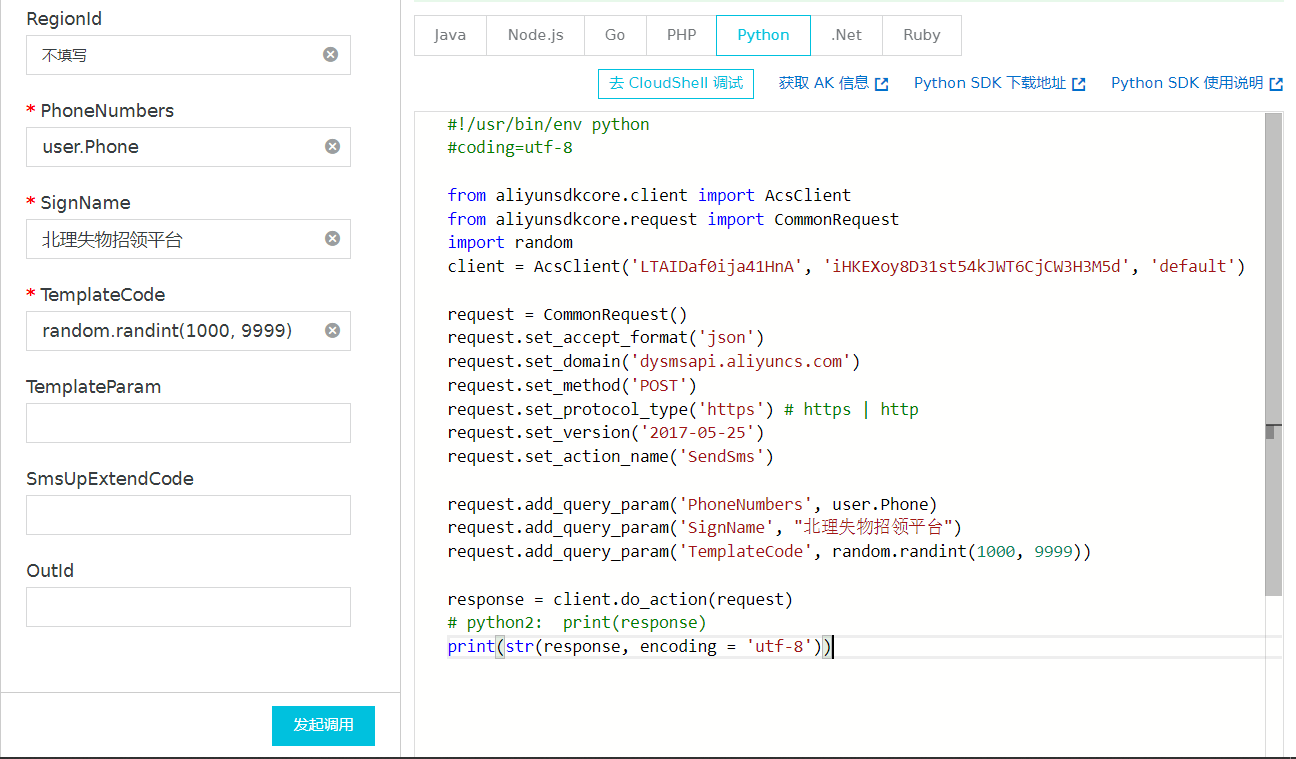
用户使用网站链接进入平台。

## 软件接口

阿里云短信服务接口：



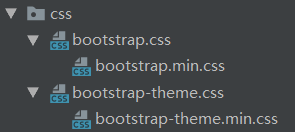




阿里云邮件服务接口：



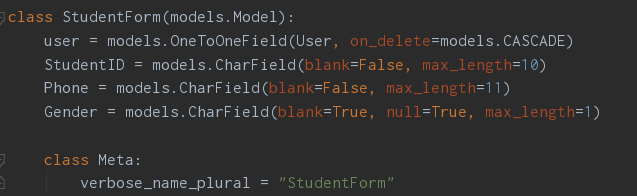
Bootstrap4镜像：

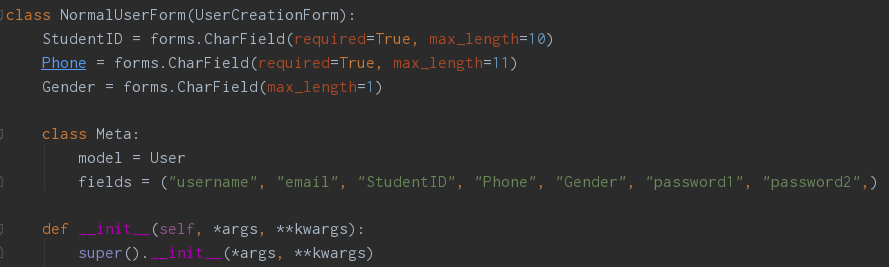


## 内部接口说明

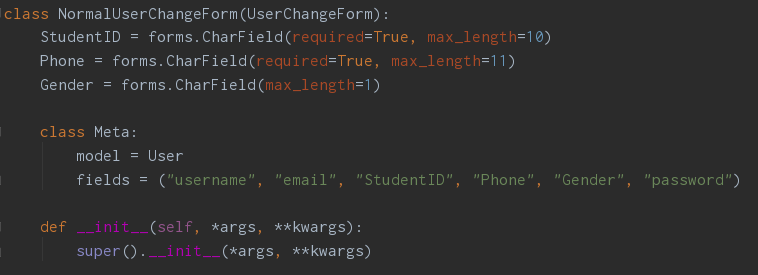
## 用户数据接口

**后端定义：**

1. 存储格式规定于models.py中，名为StudentForm，并部分继承于Django内置的models模块。
2. 创建格式规定于forms.py中，名为NormalUserForm，并部分继承于Django内置的auth.forms模块中的UserCreationForm。



1. 修改格式式规定于forms.py中，名为NormalUserChangeForm，并部分继承于Django内置的auth.forms模块中的UserChangeForm。



1. 以上三种表在views.py中调用，其中(1)用于创建、保存位于后台的用户信息，(2)(3)用于在前段生成对应表单或以该形式接受前端发送的数据。

**前端调用：**

注册表单、信息修改表单、密码修改表单在views.py中分别命名为reg\_form、edit\_form、password\_form，包含于content字典中，使用render()函数返回前端。在前端html文件中使用Django标签调用：{{ reg\_form }}、{{ edit\_form }}、{{ password\_form }}。

用户对象在后端使用request.user获取，包含于content字典中，使用render()函数返回前端。在前端html文件中使用Django标签调用获取部分属性，例：{{ user.username }}可获取用户名。

## 物品数据接口

**后端定义：**

1. 存储格式规定于models.py中，名为LostItemData和FoundItemData，并部分继承于Django内置的models模块。



1. 上表在views.py中调用，用于创建、保存位于数据库中的物品信息。

**前端调用：**

在views.py中导入LostItemData和FoundItemData，使用.objects.all()方法获得数据库中的条目信息。将其赋值给名为item的变量，包含于content字典中，使用render()函数返回前端，即可在前端htnl文件中使用Django标签获取相应信息，例：{{ item.item\_item\_publisher }}获取该物品的发布者。

## 属性

## **友好性**

网站使用了一个顶部导航栏，一个左侧边栏，保证了网站的所有功能均可以2步及以下的操作部步数调用，操作十分简单便捷，一目了然。

## 可维护性

1. 网站建立了详细明确的数据字典，明确了每个实体的名称、属性，便于维护和拓展。
2. 后端代码有详细的功能注释，便于阅读和修改。

## 可转移/换性

1. 网站使用的Django框架标签生成的表单、数据极易转换为对应html语言，迁移至其他环境或框架开发；
2. 网站使用的数据库SQLite为常用的数据库格式，稳定快捷，与另一大众数据库MySQL可互相兼容，可快捷导出sql文件并由别的数据库软件读取。

# 数据字典

## 用户信息

注：(1) 以下大部分属性为user对象的属性，可在views.py中直接调用，前端调用方法参见4.2.1；password1、password2仅属于UserCreationForm和NormalUserForm。

1. 三个password属性经过加密，无法使用Django标签在前端直接调用，但是可以在后端进行修改。

|  |  |  |
| --- | --- | --- |
| 属性 | 约束 | 释义 |
| username | 1. 20个字符或更少。 2. [包含字母，数字和仅有的@/./+/-/\_符号。](mailto:包含字母，数字和仅有的@/./+/-/_符号。) 3. 不能于已存在的用户名相同。 4. 必填。 | 用户名，用户在网站上的标识，用于显示给他人及登录用。 |
| password | 1. 不能与其他个人信息太相似。 2. 包含至少 8 个字符。 3. 不能是常见密码。 4. 不能全部为数字。 5. 必填。 | 密码，登录时使用。 |
| password1 | 1. 同password的4条。 2. 注册时必填。 | 密码，注册时使用。 |
| password2 | 1. 同password的4条。 2. 与password1相同。 3. 注册时必填。 | 密码，注册时二次确认用。 |
| emai | 1. 150个字符或更少。 2. 符合邮箱格式。 | 邮箱，用于发布消息收到互动时向用户发送通知邮件。 |
| Phone | 1. 11个字符。 2. 数字。 | 手机号，用于注册时发送验证短信、发布消息收到互动时向用户发送通知短信。 |
| Gender | 1. 1个字符。 2. “男”或“女”。 | 性别。 |

## **物品信息**

注：以下属性为LostItemData和FoundItemData中objects对象的属性。

|  |  |  |
| --- | --- | --- |
| 属性 | 约束 | 释义 |
| item\_name | 1. 20个字符或更少。 2. 必填。 | 物品名称。 |
| item\_description | 1. 50个字符或更少。 | 物品描述。 |
| item\_type | 1. 在校园卡、书籍类、学习用品、生活用品、电子产品、贵重物品、其他7个选项中选择 | 物品类型。 |
| item\_time | 1. 选择年-月-日。 2. 不能选择未来的时间。 | 丢失/捡到物品的时间。 |
| item\_location | 1. 50个字符或更少。 2. 以校区-建筑-房间形式输入（不强制要求）。 | 丢失/捡到物品的地点。 |
| item\_status | 1. True或False。 2. 用户不能输入，发布时默认为False，物品归还后置为True。 | 物品状态，False代表未寻回，True代表已寻回。 |
| item\_publisher | 1. 20个字符或更少。 2. 用户不能输入，发布时由系统自动填入发布者的username。 | 物品信息的发布者。 |