

3-2-2-3、后端逻辑

- 接口设计
- 接口定义
- 注册短信验证码路由
- 验证码实现
- 测试运行

接口设计

1.请求方式

选项	方案
请求方法	GET
请求地址	sms_codes/{?P<mobile>1[3-9]\d{9}}/

2.请求参数：路径参数

参数名	类型	是否必传	说明
mobile	string	是	手机号
image_code	string	是	图形验证码
uuid	string	是	唯一编号

3.响应结果：JSON

字段	说明
code	状态码。200表示成功；400代表失败
errmsg	错误信息

接口定义

在 verifications 子应用的 views 视图，创建 SMSCodeView

```
1 class SMSCodeView(View):
2     """短信验证码"""
3
4     def get(self, request, mobile):
5         """
6         :param request: 请求对象
7         :param mobile: 手机号
8         :return: JSON
9         """
10        pass
```

注册短信验证码路由

在 `verifications` 子应用的 `urls.py` 路由，添加如下代码

```
1 urlpatterns = [
2     ...
3     re_path(r'^sms_codes/(?P<mobile>1[3-9]\d{9})/$', SMSCodeView.as_view()) #
    短信验证码路由（正则表达式）
4     ...
5 ]
```

验证码实现

```

1 class SMSCodeView(View):
2     """短信验证码"""
3
4     def get(self, request, mobile):
5         """
6         :param request: 请求对象
7         :param mobile: 手机号
8         :return: JSON
9         """
10        # 1、接收参数
11        image_code_client = request.GET.get('img_code') # 获取验证码
12        uuid = request.GET.get('uid') # 获取uuid
13
14        # 2、校验参数
15        if not all([image_code_client, uuid]):
16            return JsonResponse({'code': 400, 'errmsg': '缺少必传参数'})
17
18        # 3、创建连接到redis的对象
19        redis_conn = redis.Redis(host='127.0.0.1', port=6379, db=0)
20        # 3-1、提取图形验证码
21        image_code_server = redis_conn.get('img:{uid}'.format(uid=uuid))
22        # 3-2、判断redis中是否后该图形验证码信息
23        if image_code_server is None:
24            # 图形验证码过期或者不存在
25            return JsonResponse({'code': 400, 'errmsg': '图形验证码失效'})
26
27        # 3-3、如果有图形验证码，那么删除图形验证码，避免恶意测试图形验证码
28        try:
29            redis_conn.delete(f'img:{uid}')
30        except Exception as e:
31            logger.error(e)
32        # 3-4、对比图形验证码
33        image_code_server = image_code_server.decode() # bytes转字符串
34        if image_code_client.lower() != image_code_server.lower(): # 转小写后
35            # 比较
36            return JsonResponse({'code': 400, 'errmsg': '输入图形验证码有误'})
37
38        # 4、生成短信验证码：生成6位数验证码
39        sms_code = '%06d' % random.randint(0, 999999)
40        logger.info(f'短信验证码是: {sms_code}')
41        # 5、保存短信验证码
42        redis_conn.setex(f'sms:{mobile}', 300, sms_code)
43        # 6、发送短信验证码
44        CCP().send_template_sms(mobile, [sms_code, 5], 1)
45
46        # 7、响应结果
47        return JsonResponse({'code': 0, 'errmsg': '发送短信成功'})

```

测试运行



商品美 · 种类多 · 欢迎光临



用户注册

[登录](#)

用户名:

密码:

确认密码:

手机号:

图形验证码:

短信验证码: [获取短信验证码](#)

请填写短信验证码

☐ 同意“美多商城用户使用协议”

注册

[关于我们](#) | [联系我们](#) | [招聘人才](#) | [友情链接](#)

Copyright © 2016 北京美多商业股份有限公司 All Rights Reserved

电话: 010-****888 京ICP备*****8号

