

“微信抢票应用”测试报告

魏宇轩 软硕192 2019214497

测试内容

- 单元测试
- 功能测试
- 性能测试

实验平台

- Ubuntu 16.0.4
- Python 3.7.0
- Django 1.9.13
- Django Server

code review

\$PROJECT_ROOT

Folder/File	Description
adminpage/	后台管理Application
codex/	自定义视图和异常类
static/	静态网页资源（html/css/javascript）
templates/	网页模板
userpage/	活动票务展现Application
wechat/	微信抢票Application
WeChatTicket/	项目主Application
configs.json	setting配置文件
manage.py	Django项目入口
requirements.txt	项目安装依赖文件

WeChatTicket/

项目主入口App，由Django框架自动生成，修改 `urls.py` 配置路由

File	Description
__init__.py	Python包文件
settings.py	配置文件
urls.py	路由
views.py	视图
wsgi.py	WSGI服务器入口

wechat/

微信抢票App，定义了用户、活动、票务的数据模型，基于微信接口开发了抢票相关功能

File	Description
management/	自定义微信相关Django command命令目录
migrations/	数据库迁移文件目录（自动生成）
__init__.py	Python包文件
admin.py	后台注册
apps.py	App配置
handlers.py	微信功能接口
models.py	数据库对象模型
settings.py	配置文件
views.py	视图
wrapper.py	微信接口底层封装

userpage/

活动票务展现App

File	Description
__init__.py	Python包文件
admin.py	后台注册
apps.py	App配置
models.py	空
urls.py	路由
views.py	视图

userpage/

后台管理App

File	Description
__init__.py	Python包文件
admin.py	后台注册
apps.py	App配置
decorator.py	装饰器
models.py	空
urls.py	路由
views.py	视图

生成测试数据

数据库配置

库名	wechat_ticket
username	root
password	o443W586A2
ip	127.0.0.1
port	3306

生成数据表

利用Django框架自动按照 `./wecaht/models.py` 生成数据表 `wechat_activity`, `wechat_ticket`, `wechat_user`

```
mysql> show tables;
+-----+
| Tables_in_wechat_ticket |
+-----+
| auth_group               |
| auth_group_permissions   |
| auth_permission          |
| auth_user                |
| auth_user_groups         |
| auth_user_user_permissions |
| django_admin_log         |
| django_content_type      |
| django_migrations        |
| django_session           |
| wechat_activity          |
| wechat_ticket            |
| wechat_user              |
+-----+
13 rows in set (0.00 sec)
```

生成测试数据

- 用 `python manager.py createsuperuser` 命令创建管理员超级用户
 - username: root
 - password: !root123
- 使用Python脚本向MySQL插入测试用户数据

```
"""
insert test data into database for unit test and function test
"""

import MySQLdb

db = MySQLdb.connect('127.0.0.1', 'root', 'o443W586A2', 'wechat_ticket')

cursor = db.cursor()

sql = """INSERT INTO wechat_user(open_id, to_active) VALUES ('weiyx', 1)"""

try:
    cursor.execute(sql)
    db.commit()
    print('success~~')
except:
    db.rollback()
    print('fail~~')

db.close()
```

```
mysql> select * from wechat_user;
+-----+-----+-----+-----+
| id | open_id | to_active | system_user_id |
+-----+-----+-----+-----+
| 5 | weiyx | 1 | NULL |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

单元测试

测试范围

adminpage/views.py 中的各个视图是否能按照 adminpage/urls.py 里的路径正确访问

- ▼ views.py
 - ▶ ActivityCheckin(APIView)
 - ▶ ActivityCreate(APIView)
 - ▶ ActivityDelete(APIView)
 - ▶ ActivityDetail(APIView)
 - ▶ ActivityList(APIView)
 - ▶ ActivityMenu(APIView)
 - ▶ AdminLoginView(APIView)
 - ▶ AdminLogoutView(APIView)
 - ▶ ImageUpload(APIView)

测试代码

```
class UrlUnitTest(TestCase):
    def test_login_url(self):
        response = self.client.get('/a/login/')
        self.assertEqual(response.status_code, 200)

    def test_activity_list_url(self):
        response = self.client.get('/a/activity/list')
        self.assertEqual(response.status_code, 200)

    def test_acitivity_detail_url(self):
        response = self.client.get('/a/activity/detail?create=1')
        self.assertEqual(response.status_code, 200)

    def test_acitivity_menu_url(self):
        response = self.client.get('/a/activity/menu')
        self.assertEqual(response.status_code, 200)

    def test_activity_checkin_url(self):
        response = self.client.get('/a/activity/checkin')
        self.assertEqual(response.status_code, 200)
```

测试结果

登录、活动列表显示、新建活动、活动菜单、活动检票等页面可以正确访问

功能测试

测试范围

```
urlpatterns = [  
    url(r'^login/$', AdminLoginView.as_view()),  
    url(r'^logout/$', AdminLogoutView.as_view()),  
    url(r'^activity/list/$', ActivityList.as_view()),  
    url(r'^activity/detail/$', ActivityDetail.as_view()),  
    url(r'^activity/create/$', ActivityCreate.as_view()),  
    url(r'^activity/delete/$', ActivityDelete.as_view()),  
    url(r'^activity/menu/$', ActivityMenu.as_view()),  
    url(r'^image/upload/$', ImageUpload.as_view()),  
    url(r'^activity/checkin/$', ActivityCheckin.as_view())  
]
```

管理员登录

测试用例

1. 正确用户名 + 正确密码
 2. 错误用户名
 3. 正确用户名 + 错误密码
- 注：在正确登录的测试用例中保存cookie信息到本地文件，之后测试需要登录的页面时从本地文件加载cookie即可

测试代码

```
class AdminLoginViewTest(FunctionTestWrapper):  
    def test_valid_login(self):  
        self.browser.get('http://127.0.0.1:8000/a/login')  
        inputUserName = self.browser.find_element_by_id('inputUsername')  
        inputPassword = self.browser.find_element_by_id('inputPassword')  
        loginnow = self.browser.find_element_by_id('loginnow')  
  
        inputUserName.clear()  
        inputUserName.send_keys('root')  
  
        inputPassword.clear()  
        inputPassword.send_keys('!root123')  
  
        loginnow.click()  
        time.sleep(2)  
  
        self.assertEqual(self.browser.title, '活动列表 - 紫荆之声') # check if jump to  
        activity list page
```

```

cookies = self.browser.get_cookies()

with open('admin_cookies.json', 'w') as f:
    json.dump(cookies, f)

def test_invalid_username(self):
    self.browser.get('http://127.0.0.1:8000/a/login')
    inputUserName = self.browser.find_element_by_id('inputUsername')
    inputPassword = self.browser.find_element_by_id('inputPassword')
    loginnow = self.browser.find_element_by_id('loginnow')

    inputUserName.clear()
    inputUserName.send_keys('weiyx')

    inputPassword.clear()
    inputPassword.send_keys('!root123')

    loginnow.click()

    self.assertEqual(self.browser.find_element_by_id('alert').text, '用户名或密码错误')

def test_wrong_password(self):
    self.browser.get('http://127.0.0.1:8000/a/login')
    inputUserName = self.browser.find_element_by_id('inputUsername')
    inputPassword = self.browser.find_element_by_id('inputPassword')
    loginnow = self.browser.find_element_by_id('loginnow')

    inputUserName.clear()
    inputUserName.send_keys('root')

    inputPassword.clear()
    inputPassword.send_keys('root')

    loginnow.click()

    self.assertEqual(self.browser.find_element_by_id('alert').text, '用户名或密码错误')

```

测试结果

```
-----
Ran 3 tests in 39.549s
```

```
OK
```

创建活动

测试用例

根据等价类划分法构造表单测试用例，分别测试“提交”“暂存”“重置”功能，重点各种非法输入

测试代码

```

class ActivityListTest(FunctionTestWrapper):
    def setUp(self) -> None:
        self.browser = webdriver.Firefox()
        self.browser.get('http://127.0.0.1:8000/a/activity/detail?create=1')

        self.browser.delete_all_cookies()
        with open('admin_cookies.json', 'r', encoding='utf-8') as f:
            cookies = json.load(f)
            for cookie in cookies:
                self.browser.add_cookie({
                    'name': cookie['name'],
                    'value': cookie['value']
                })

        self.browser.get('http://127.0.0.1:8000/a/activity/detail?create=1')
        time.sleep(.5)

        inputName = self.browser.find_element_by_id('input-name')
        inputName.clear()
        inputName.send_keys('faust')
        time.sleep(.5)

        inputKey = self.browser.find_element_by_id('input-key')
        inputKey.clear()
        inputKey.send_keys('faust')
        time.sleep(.5)

        inputPlace = self.browser.find_element_by_id('input-place')
        inputPlace.clear()
        inputPlace.send_keys('assembly hall')
        time.sleep(.5)

        inputDescription = self.browser.find_element_by_id('input-description')
        inputDescription.clear()
        inputDescription.send_keys('very good')
        time.sleep(.5)

        inputPicUrl = self.browser.find_element_by_id('input-pic_url')
        inputPicUrl.clear()
        inputPicUrl.send_keys(''https://img-blog.csdnimg.cn/20190504232138969.PNG?x-oss-
process=image/watermark,
type_ZmFuZ3poZW5naGVpdGk,shadow_10,text_aHR0cHM6Ly9ibG9nLmNzZG4ubmV0L2RhX2thb19sYQ==,size_1
6,color_FFFFFFFF,
t_70'')
        time.sleep(.5)

        inputTotalTickets = self.browser.find_element_by_id('input-total_tickets')
        inputTotalTickets.clear()
        inputTotalTickets.send_keys('1000')
        time.sleep(.5)

        self.timeList = list()

```



```

        for field0 in ('start', 'end', 'book-start', 'book-end'):
            tmpList = list()
            for field1 in ('year', 'month', 'day', 'hour', 'minute'):
                tmpList.append(self.browser.find_element_by_id('input-{}-{}'.format(field0,
field1)))
            self.timeList.append(tmpList)
            self.inputTuple = ('2019', '10', '2', '0', '0', '2019', '10', '2', '1', '0',
'2019', '10', '1', '0', '0',
                                '2019', '10', '1', '12', '0')
            idx = 0
            for tmpList in self.timeList:
                for val in tmpList:
                    val.clear()
                    val.send_keys(self.inputTuple[idx])
                    idx += 1
                    time.sleep(.5)

    def test_save(self):
        saveBtn = self.browser.find_element_by_id('saveBtn')
        saveBtn.click()

        idx = 0
        for field0 in ('start', 'end', 'book-start', 'book-end'):
            for field1 in ('year', 'month', 'day', 'hour', 'minute'):
                self.assertEqual(self.browser.find_element_by_id('input-{}-
{}').format(field0, field1)).get_attribute('value'),
                                self.inputTuple[idx])
                idx += 1

    def test_submit(self):
        publishBtn = self.browser.find_element_by_id('publishBtn')
        publishBtn.click()
        time.sleep(1)

        self.assertIsNotNone(self.browser.find_element_by_id('resultHolder'))

    def test_reset(self):
        resetBtn = self.browser.find_element_by_id('resetBtn')
        resetBtn.click()
        time.sleep(.5)

        self.assertEqual(self.browser.find_element_by_id('input-
name').get_attribute('value'), '')

```

测试结果

1. 暂存或重置表单后部分字段被置为"disabled"无法修改
2. 提交表单后“返回修改”回到填写表单页面部分字段被置为"disabled"无法修改
3. 提交表单后“返回修改”回到填写表单页面部分字段提交结果无效（显示字段xx未填写）
4. 提交表单后“返回修改”回到填写表单页面后“重置”不起作用
5. 时间约束不正确（活动开始/结束时间与抢票开始/结束时间的关系）
6. 通过/a/activity/list的详情页进入创建活动页面后“提交”总是失败
7. 通过/a/activity/list的详情页进入创建活动页面后“重置”不起作用

- 注：之后的几项测试仅展示测试结果，测试步骤略

管理员登出

可以正确登出

活动列表

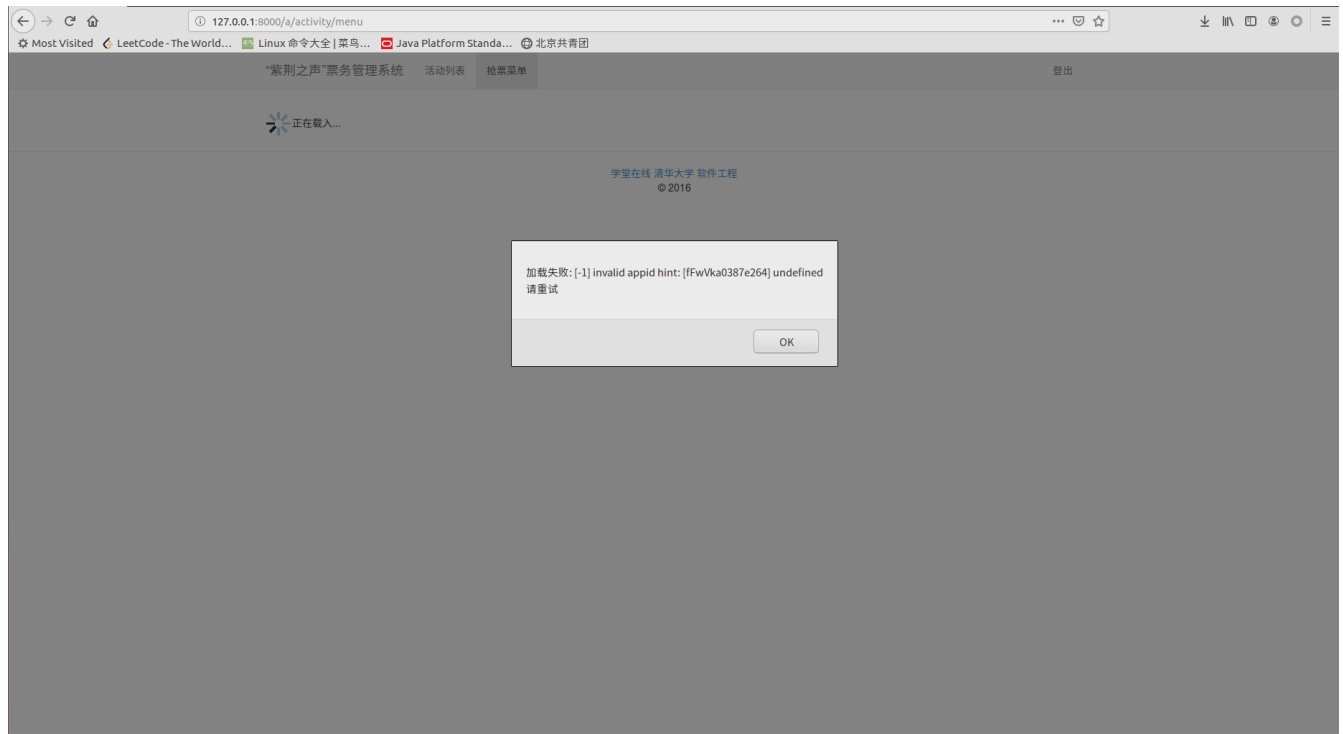
可以正确查看活动

删除活动

可以正确删除活动

抢票菜单

微信接口失效，无法抢票



图片上传

上传成功后“暂存”仍提示"please enter a url"

检票

检票显示二维码页面js出错

“紫荆之声”票务系统 - 检票

test2

活动时间：2019年11月01日 周五 00:00 - 01:00

扫描二维码

Inspector Console Debugger Style Editor Performance Memory Network Storage Accessibility

Filter output

SyntaxError: missing] after element list [Learn More](#)
note: | opened at line 14, column 0

SyntaxError: missing] after element list [Learn More](#)
note: | opened at line 14, column 0

activity_checkin.js:14:8
activity_checkin.js:14

activity_checkin.js:14:8
activity_checkin.js:14

>>