

《Outfit of ToDay》项目测试报告

杜婉晴 2021011824 陈植 2021011798 李昊轩 2021012900 苟芳菲 2021011837

此报告由功能测试和性能测试两部分组成，测试环境均为助教提供的服务器，与交付产品一致。我们保证数据的真实性。

一. 功能测试

1. 单元测试：采用 Python 的标准模块unittest

(1) 00TD_backend/login/tests.py

待测 API	API 功能	请求类型	测试用例
edit_info	修改用户信息	PATCH	数据正确 数据错误：性别填写格式有误 数据错误：年龄为负数 数据错误：电话号码长度有误 数据错误：地点不存在
user	获取用户信息	GET	请求无需发送数据
get_weather	获取天气	GET	请求无需发送数据

(2) 00TD_backend/00TD_closet/tests.py

待测 API	API 功能	请求类型	测试用例
get_clothes	获取衣服列表信息	GET	请求无需发送数据
add_outfit	将衣柜内的衣服添加至搭配栏	POST	数据正确：添加上衣 1 数据正确：添加裤子 数据正确：添加鞋子 数据正确：添加帽子 数据错误：发送的 clothes id 为负数
remove_outfit	移除搭配栏中的指定衣服	POST	数据正确：移除帽子 数据错误：发送的 clothes id 为

			负数
get_outfit	获取搭配栏内的衣服信息	GET	请求无需发送数据
score	给穿搭评分	POST	请求无需发送数据
generate	生成推荐穿搭	POST	请求无需发送数据

使用 coverage 工具统计覆盖率

在 00TD_backend 路径下运行以下命令，

1. `coverage run manage.py test`
2. `coverage report`
3. `coverage html`

运行上述命令后，htmlcov 目录下包含 HTML 格式的覆盖率报告，可通过浏览器打开 `htmlcov/index.html` 来查看详细的覆盖率信息。

覆盖率如下截图：

Coverage report: 68%

coverage.py v6.4.4, created at 2024-01-04 14:25 +0800

Module	statements	missing	excluded	coverage
OOTD_closet__init__.py	0	0	0	100%
OOTD_closet\admin.py	5	0	0	100%
OOTD_closet\apps.py	4	0	0	100%
OOTD_closet\migrations\0001_initial.py	6	0	0	100%
OOTD_closet\migrations__init__.py	0	0	0	100%
OOTD_closet\models.py	36	6	0	83%
OOTD_closet\tests.py	90	1	0	99%
OOTD_closet.urls.py	3	0	0	100%
OOTD_closet\views.py	336	158	0	53%
login__init__.py	0	0	0	100%
login\admin.py	3	0	0	100%
login\apps.py	4	0	0	100%
login\controllers.py	42	29	0	31%
login\migrations\0001_initial.py	6	0	0	100%
login\migrations__init__.py	0	0	0	100%
login\models.py	35	6	0	83%
login\tests.py	46	1	0	98%
login.urls.py	5	0	0	100%
login\views.py	139	54	0	61%
manage.py	12	2	0	83%
ootd__init__.py	0	0	0	100%
ootd\settings.py	25	0	0	100%
ootd.urls.py	7	1	0	86%
posting__init__.py	0	0	0	100%
posting\admin.py	1	0	0	100%
posting\apps.py	4	0	0	100%
posting\forms.py	12	0	0	100%
posting\migrations\0001_initial.py	6	0	0	100%
posting\migrations__init__.py	0	0	0	100%
posting\models.py	23	0	0	100%
posting\serializers.py	23	0	0	100%
posting\tests.py	8	0	0	100%
posting.urls.py	3	0	0	100%
posting\views.py	112	89	0	21%
tests__init__.py	0	0	0	100%
tests\test_api.py	26	1	0	96%
utils\fashion_compatibility_mcn\diagnosis.py	123	62	0	50%
utils\fashion_compatibility_mcn\model.py	139	44	0	68%
utils\fashion_compatibility_mcn\resnet.py	134	36	0	73%
utils\fashion_compatibility_mcn\revision.py	93	4	0	96%
utils\jwt.py	47	6	0	87%
Total	1558	500	0	68%

coverage.py v6.4.4, created at 2024-01-04 14:25 +0800

2. 端到端测试

页面	功能		测试结果
登录界面	登录获取用户信息		通过
我的衣柜	衣柜	添加衣服	通过
		编辑衣服	通过
		删除衣服	通过
	搭配栏	拖拽添加衣服	通过
		删除衣服	通过
		同类衣服自动替换	通过
	穿搭评分		通过
	生成推荐穿搭		通过
	一键替换穿搭		通过
	展示天气		通过
个人中心	个人信息显示		通过
	搭配评分折线图显示		通过
	最常穿衣服饼状图显示		通过
资料编辑	上传头像		通过
	修改昵称、手机号、地点、年龄、性别		通过
发帖	添加图片		通过
	添加标题、内容描述		通过
	展示评分		通过
	展示天气		通过
	分享天气		通过
社区	全部	浏览帖子	通过
		点赞	通过
		收藏	通过
		评论	通过
	收藏栏	已收藏帖子展示正常	通过
	我的	显示当前用户的帖子	通过

二. 性能测试

本项目性能测试通过划分子系统分为三个部分，测试服务接口在高并发访问下是否还能正常工作、响应速度如何。以下所有测试中，均没有出现错误，即服务接口总是被正确调用的。值得一提的是，在分享社区接口测试中，我们根据现实情况模拟了两种高并发场景。一是大量用户在短时间内创建贴文并浏览，二是大量用户在长时间内不断浏览社区和贴文。在两种测试中，系统的表现都令人满意。

1. 虚拟衣柜相关接口测试

模拟用户的行为如下：

- （1）编辑衣物信息（停顿 1-2 秒）
- （2）上传衣物图片和信息
- （3）重复前两步，直到已经上传 8 件衣物
- （4）浏览虚拟衣柜

当在 5 秒内产生 100 个线程来对服务器进行访问时，得到的结果如下：

Label	Samples	Average	Median	90% Line	95% Line	Min	Max
add clothes	800	115	107	164	184	62	443
get clothes	100	85	78	103	124	71	151
TOTAL	800	115	107	164	184	62	443

可见，各项请求的响应时长的最大值都在 0.5 秒以内，响应十分迅速和稳定，满足要求。

2. 穿搭推荐接口测试

经过大量测试，部署在服务器上的推荐算法的响应时长在 0.5 秒左右，基本满足要求。但由于推荐算法是基于一个开源模型实现的，在服务器资源有限的情况下并不能很好的满足并发请求要求。当 5 个线程同时请求时，平均响应时长会来到 2 秒左右。

3. 分享社区相关接口测试

模拟用户的行为如下：

- （1）创建一个帖子，帖子包含三张图片
- （2）浏览贴文列表
- （3）翻阅（随机停顿 1-4 秒）
- （4）打开一个帖子浏览详情

当在 5 秒内产生 200 个线程来对服务器进行访问时，得到的结果如下：

Label	Samples	Average	Median	90% Line	95% Line	Min	Max
create post	200	927	897	1746	1863	94	1997
upload image	600	1272	1393	1969	1994	96	2060
user posts	200	1014	813	1949	1989	56	2052
post detail	200	883	510	1918	1963	51	2025
TOTAL	1200	1107	1124	1951	1988	51	2060

可见，各项请求的平均响应时长在 1 秒所有，最大响应时长在 2 秒左右，满足要求。

接下来打开 100 个线程，每个线程重复浏览贴文的操作，持续 2 分钟，结果如下：

Label	Samples	Average	Median	90% Line	95% Line	Min	Max
user posts	1611	2240	2288	2574	2675	144	3184
post detail	1584	2094	2125	2392	2482	52	2661
TOTAL	3195	2168	2202	2499	2587	52	3184

可见，此时服务的平均响应时长在 2 秒左右，95%的响应时长在 2.5 秒左右，符合要求。