**聚票网—基于kubernetes的分布式系统架构实现**

**测试用例**

**版本1.0**

|  |  |  |  |
| --- | --- | --- | --- |
| **修订历史** | | | |
| 日期 | 版本 | 说明 | 作者 |
| 2018/07/26 | 1.0 | 第一次迭代 | 潘子奕、  裘鹏泽、  徐天强、  应邦豪 |
| 2018/09/02 | 2.0 | 第二次迭代 |
|  |  |  |
|  |  |  |

目录

1. 测试用例1

1.1 第一次测试2

键入章标题(第 3 级)3

键入章级(第 1 级)4

键入章级(第 2 级)5

键入章标题(第 3 级)6

**1.测试用例**

2.1 第一次测试

用例编号：SIGN\_UP\_test1 版本号：1.0

模块名称：signup 用例级别：集成测试

测试时间：2018年7月26日 测试人员：徐天强

测试目的：测试是否能够正常注册

预置条件：无

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在首页边栏点击“Sign in/up”按钮 | 输入用户名，密码，邮箱与验证码 | 页面跳转到登录界面 | 符合期望 |
| 2 | 点击“Sign ip” | 页面跳转到注册界面 |
| 3 | 填入合格的用户名，密码，邮箱，验证码 | 显示已经发送激活邮件到填写的邮箱 |
| 4 | 在激活邮箱里点击链接 | 跳转到激活成功界面 |
| 5 | 使用已激活的用户名与密码进行登录 | 登录成功 |

用例编号：SIGN\_IN\_test1 版本号：1.0

模块名称：signin 用例级别：集成测试

测试时间：2018年7月26日 测试人员：徐天强

测试目的：检测对已注册的用户是否能够正常登录

预置条件：数据库中已经存入了相关用户的用户名与密码等信息

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在首页边栏点击“Sign in/up”按钮 | 输入用户名，密码与验证码 | 页面跳转到登录界面 | 符合期望 |
| 2 | 填入相匹配的用户名与密码，并依照验证码图片填写验证码 | 无 |
| 3 | 点击“LOG IN”按钮 | 显示登录成功，并且网站切换到登录后的状态 |

用例编号：USERINFO\_test1 版本号：1.0

模块名称：userinfo 用例级别：集成测试

测试时间：2018年7月26日 测试人员：徐天强

测试目的：查看用户的相关信息，并修改地址、手机号码等信息

预置条件：系统中已经有该用户的基本信息，并且成功登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在边栏点击“Account”按钮，待嵌套的按钮显示后，点击“UserInfo”按钮 | 输入手机号码、昵称、账户余额、地址信息 | 页面跳转到用户信息页面 | 符合期望 |
| 2 | 点击页面右侧的带有笔icon的按钮 | 用户的手机号码、昵称等信息变成可修改的状态 |
| 3 | 填写 手机、昵称、账户余额、地址信息 | 用户的相关信息在页面上显示改变 |
| 4 | 点击页面右侧的橘色的带有保存icon的按钮，进行保存 |  | 用户的信息被保存 |  |
| 5 | 点击边栏任意按钮，跳转到其他页面，再点击“UserInfo”返回到用户信息页面 | 用户信息页面显示保存后的信息 |

用例编号：Cart\_DetailModal\_test1 版本号：1.0

模块名称：detailmodal+ cart 用例级别：集成测试

测试时间：2018年7月26日 测试人员：徐天强

测试目的：在相关页面的票品卡片中点击橙色”ADD“按钮，加入购物车

预置条件：用户已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在边栏点击“Home”按钮，home页面下的票品显示之后，点击橙色的“···”按钮，进入票品卡片 | 无 | 页面跳转到home页面，在点击“···”按钮之后，进入票品卡片页面 | 符合期望 |
| 2 | 在票品卡片中选择日期与相应票价的票品数量 | 页面上对选择的票品日期与相应价格的票品数量进行显示 |
| 3 | 点击橙色“ADD”按钮，加入购物车 | 网站通过alert的方式，提示“成功！“ |
| 4 | 退出票品卡片界面，在左侧边栏中点击“Cart”按钮 |  | 跳转到cart页面 |  |
| 5 | 在购物车页面查看购物车中的物品 | 购物车内分页显示用户加入购物车的票品 |

用例编号：Order\_Cart\_test1 版本号：1.0

模块名称：order + cart 用例级别：集成测试

测试时间：2018年7月26日 测试人员：徐天强

测试目的：在相关页面的票品卡片中点击橙色”ADD“按钮，加入购物车

预置条件：用户已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在左侧边栏点击“Cart”按钮 | 无 | 页面跳转到购物车页面 | 符合期望 |
| 2 | 在购物车页面，批量选择购物车中的票品 | 页面显示相关的购物车商品被选中 |
| 3 | 点击页面下方的“结算”按钮 | 跳转到“订单信息确认页面” |
| 4 | 在“orderconfirm”页面下点击“提交”按钮 | 跳转到“支付确认页面” |
| 5 | 在“payconfirm”页面下点击“支付”按钮 | 跳转到支付情况反馈的页面 |
| 6 | 在页面左侧边栏中点击 “Account”按钮，待嵌套的按钮显示后，点击“Order”按钮 |  | 进入用户的订单页面，显示刚刚支付的订单信息，如果支付成功则订单的状态改为“待发货” |  |

用例编号：comment\_test1 版本号：2.0

模块名称：comment 用例级别：集成测试

测试时间：2018年9月2日 测试人员：潘子奕

测试目的：在票品底部可以发表评论并修改或删除自己已发表评论

预置条件：用户已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在票品底部输入评论内容并点击评论 | 评论内容或修改内容 | 评论内容出现在评论区 | 符合期望 |
| 2 | 用户点击评论修改按钮并修改评论 | 评论内容被修改成功 |
| 3 | 用户点击评论删除按钮 | 评论删除成功 |

用例编号：collection\_test1 版本号：2.0

模块名称：collection 用例级别：集成测试

测试时间：2018年9月2日 测试人员：潘子奕

测试目的：在票品底部可以发表评论并修改或删除自己已发表评论

预置条件：用户已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在票品首页点击星形按钮 | 无 | 页面allert收藏成功 | 符合期望 |
| 2 | 用户进去收藏夹页面 | 用户成功发现自己的票品已经被收藏 |
| 3 | 用户点击票品分享按钮 | 弹出对应平台的二维码分享码 |

用例编号：collection\_test1 版本号：2.0

模块名称：collection 用例级别：集成测试

测试时间：2018年9月2日 测试人员：潘子奕

测试目的：用户可以收藏并分享自己喜欢的票品

预置条件：用户已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 用户在票品首页点击星形按钮 | 无 | 页面allert收藏成功 | 符合期望 |
| 2 | 用户进去收藏夹页面 | 用户成功发现自己的票品已经被收藏 |
| 3 | 用户点击票品分享按钮 | 弹出对应平台的二维码分享码 |

用例编号：manager\_ticket\_test1 版本号：2.0

模块名称：manager 用例级别：集成测试

测试时间：2018年9月2日 测试人员：潘子奕

测试目的：测试管理员的票品管理相关功能

预置条件: 管理员已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 管理员进入票品增加页面 | 无 | 页面出现表单 | 符合期望 |
| 2 | 管理员填写表单并提交 | 页面弹出成功窗口，并可以看到新票品出现 |
| 3 | 管理员进入票品修改窗口 | 页面出现表单并已票品原有信息为默认值 |
|  | 票品修改表单并提交 |  | 页面弹出成功窗口，并可以看到新票品出现 |  |
|  | 管理员点击票品删除按钮 |  | 票品删除成功 |  |

用例编号：manager\_user\_test1 版本号：2.0

模块名称：manager 用例级别：集成测试

测试时间：2018年9月2日 测试人员：潘子奕

测试目的：测试管理员的用户管理相关功能

预置条件: 管理员已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 管理员进入用户管理页面 | 无 | 页面出现所有用户的简单信息 | 符合期望 |
| 2 | 管理员单击修改按钮并对于用户信息进行修改 | 用户信息修改成功 |
| 3 | 管理员单击删除按钮 | 用户删除成功 |
|  | 管理员单击冻结按钮 |  | 用户状态变为冻结 |  |
|  | 管理员点击添加按钮，并提交相关信息 |  | 新用户添加成功 |  |

用例编号：manager\_report\_test1 版本号：2.0

模块名称：manager 用例级别：集成测试

测试时间：2018年9月2日 测试人员：潘子奕

测试目的：测试管理员的报表统计相关功能

预置条件: 管理员已经登录

测试步骤：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作步骤 | 操作内容 | 输入数据 | 期望结果 | 测试结果 |
| 1 | 管理员进入报表统计页面 | 无 | 页面出现选择界面 | 符合期望 |
| 2 | 管理员选择确定的一个日期 | 页面出现当天的报表统计情况 |
| 3 | 管理员选择确定的一周 | 页面出现该周的报表统计情况 |
|  | 管理员选择确定的一月 |  | 页面出现该月的报表统计情况 |  |
|  | 管理员选择具体的城市 |  | 页面出现筛选后的报表统计情况 |  |