测试报告

基础信息模块

basic information Module

作者

3170104705 杨淯晴

3170106317 仇盛显

3170106089 马跃洋

浙江大学计算机科学与技术学院战疫软工开发组

2020年5月

目录

[1引言 1](#_Toc41082670)

[1.1编写目的 1](#_Toc41082671)

[1.2背景 1](#_Toc41082672)

[1.2.1软件系统名称 1](#_Toc41082673)

[1.2.2主要任务 1](#_Toc41082674)

[1.2.3任务提出者 1](#_Toc41082675)

[1.2.4开发者 1](#_Toc41082676)

[1.3定义 2](#_Toc41082677)

[1.4参考资料 2](#_Toc41082678)

[2测试概要 3](#_Toc41082679)

[2.1系统功能说明 3](#_Toc41082680)

[2.2测试内容与时间安排 3](#_Toc41082681)

[3功能测试 4](#_Toc41082682)

[3.1会话管理 4](#_Toc41082683)

[3.1.1后端 4](#_Toc41082684)

[3.1.2前端 5](#_Toc41082685)

[3.2个人用户信息管理 9](#_Toc41082686)

[3.2.1后端 9](#_Toc41082687)

[3.2.2前端 10](#_Toc41082688)

[3.3商家用户信息管理 12](#_Toc41082689)

[3.3.1后端 12](#_Toc41082690)

[3.3.2前端 14](#_Toc41082691)

[3.4 后台管理系统 16](#_Toc41082692)

[4边界测试 19](#_Toc41082693)

[4.1会话管理 19](#_Toc41082694)

[4.2个人用户信息管理 20](#_Toc41082695)

[4.3商家用户信息管理 20](#_Toc41082696)

[4.4后台信息管理 21](#_Toc41082697)

[5持续集成 23](#_Toc41082698)

[6压力测试 23](#_Toc41082699)

[7分析与结论 24](#_Toc41082700)

[7.1测试情况汇总 24](#_Toc41082701)

[7.2结论 24](#_Toc41082702)

# 1引言

## 1.1编写目的

本测试报告编写目的主要是记录软件测试的结果，以及确认是否所有的功能都正常实现了，有助于完善我们编写的基础信息模块，提供更好的服务于用户。软件测试的基本内容包括基本情况分析、测试需求说明、测试策略和记录等等。软件测试的基本方法包括功能测试、边界测试、压力测试等等。本文档面向的读者包括: 系统项目开发人员、系统项目管理人员、系统测试人员等。

## 1.2背景

### 1.2.1软件系统名称

疫情监测和服务系统——基础信息模块

### 1.2.2主要任务

基础信息模块是疫情监测与服务系统的重要组成部分，包含登陆注册模块、用户信息中心模块以及后台运营人员模块，用户通过申请一个帐号来使用此系统，在用户信息中心能查看、修改个人信息，后台运营人员能修改用户的权限以及随时根据需求处理用户反馈的错误。

### 1.2.3任务提出者

本项目的任务提出者为浙江大学计算机学院王章野老师

### 1.2.4开发者

本项目的开发者为浙江大学计算机学院软件工程

基础信息模块小组成员:杨淯晴、仇盛显、马跃洋

## 1.3定义

* 功能测试(Functional Testing):也称为行为测试(Behavioral Testing)，根据产品特征、操作描述和用户方案，测试一个产品的特性和可操作性行为以确定它们满足设计需求。本地化软件的功能测试，用户验证应用程序或网站对目标用户能正确工作，使用适当的平台、浏览器和测试脚本，以保证目标用户的体验将足够好，就像应用程序是专门为该市场开发的一样。
* 边界测试(Boundary Testing):顾名思义，就是用来探测和验证代码在处理极端或偏门的情况时会发生什么，是否能够产生预计的返回结果。
* 压力测试(Stress Testing):软件压力测试是一种基本的质量保证行为，它是每一个重要软件测试 工作的一部分。软件压力测试的基本思路很简单:不是在常规条件下运行手动或自动测试，而是在 计算机数量较少或系统资源匮乏的条件下运行测试。通常要进行软件压力测试的资源包括内部内 存、CPU 可用性、磁盘空间和网络带宽等。

### 1.4参考资料

《测试报告参考 1》 提供者:王章野

《测试报告参考 2》 提供者:王章野

# 2测试概要

## 2.1系统功能说明

|  |  |  |
| --- | --- | --- |
| **功能** | **输入** | **输出** |
| 登录 | 用户ID，密码 | Session token |
| 注册 | 用户ID，密码，用户信息 | 最终状态结果 |
| 用户信息查看 | 用户名 | 用户信息 |
| 用户信息修改 | 修改条目 | 修改结果 |
|  |  |  |

## 2.2测试内容与时间安排

|  |  |  |  |
| --- | --- | --- | --- |
| **测试类型** | **测试目的** | **测试方式** | **时间排期** |
| 功能测试 | 测试各个功能能否按照预期工作 | 主要从前端直接操作 | 与开发基本并行 |
| 边界测试 | 测试程序能否对边界数据的输入正确工作 | 测试前端的边界数据与通过API直接向后端提交非法数据 | 开发基本结束后开始 |
| 压力测试 | 测试大负载下系统的性能和服务能力 | 使用工具构造常见请求向后端自动提交 | 开发与前两部分测试完成后开始 |

# 3功能测试

除用户登录、注册外，所有的操作都需要在请求头中添加token，此处不再反复说明。

## 3.1会话管理

### 3.1.1后端

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输 出** | **说明** |
| 用户登录 | username：String  password：String | username:String  ifpersonalUser:String | 符合预期 | 如果登录成功，返回用户的用户名和用户类型；如果失败，返回失败原因 |
| 个人用户注册 | username:String password:String confiredpassword:String name:String gender:String identityCardNumber:String phoneNumber:String email:String address:String healthCode:String healthStatus:String | username:String password:String name:String gender:String identityCardNumber:String phoneNumber:String email:String address:String healthCode:String healthStatus:String | 符合预期 | 如果注册成功，返回个人用户填写的用户信息；如果失败，返回失败原因 |
| 商家用户注册 | username:String password:String confiredpassword:String name:String phoneNumber:String email:String address:String businessLicense:String  corporateIdentity:String  category:String  collectionInformation:String | username:String password:String name:String phoneNumber:String email:String address:String businessLicense:String  corporateIdentity:String  category:String  collectionInformation:String | 符合预期 | 如果注册成功，返回商家用户填写的用户信息；如果失败，返回失败原因 |
| 用户登出 | None | Username:Strign | 符合预期 | 如果登出成功，返回用户的用户名；如果失败，返回失败原因 |

### 3.1.2前端

|  |  |  |
| --- | --- | --- |
| **功能** | **输入** | **测试输出** |
| 用户登录 |  |  |
| 用户登录 | 用户名不存在时 | 1589985523(1) |
| 用户登录 | 密码错误时 | 1589985550(1) |
| 个人用户注册 |  |  |
| 个人用户注册 | 当有未项时 | 1589984826(1) |
| 个人用户注册 | 当某项过长时 | 1589984870(1) |
| 个人用户注册 | 当密码和确认密码不同时 | 1589985478(1) |
| 个人用户注册 | 当健康状态填入内容不符合标准时 | 1589984956(1) |
| 个人用户注册 | 当健康码填入内容不符合标准时 | 1589985036(1) |
| 商家用户注册 |  |  |
| 商家用户注册 | 当有未填项时 | 1589985199(1) |
| 商家用户注册 | 当某项过长时 | 1589985225(1) |
| 商家用户注册 | 当密码和确认密码不同时 | 1589985330(1) |
| 用户登出 | 点击 |  |
| 自动登出 | 在线时间持续超过8分钟 |  |
| 后台登录 | 输入错误管理员账号或密码 |  |
| 后台登录 | 输入正确的管理员账号和密码 |  |

经测试，输出均符合要求

## 3.2个人用户信息管理

### 3.2.1后端

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输出** | **说明** |
| 获取用户信息 | username：String  ifpersonalUser:String | name:String gender:String identityCardNumber:String phoneNumber:String email:String address:String healthCode:String healthStatus:String | 符合预期 | 根据登录的username和用户类型返回当前个人用户的信息 |
| 修改用户姓名 | username:String  name:String | name:String | 符合预期 | 如果修改成功，返回修改后的姓名；如果失败，返回失败原因 |
| 修改用户电话 | username:String  phoneNumber:String | phoneNumber:String | 符合预期 | 如果修改成功，返回修改后的电话；如果失败，返回失败原因 |
| 修改用户邮箱 | username:String  email:String | email:String | 符合预期 | 如果修改成功，返回修改后的邮箱；如果失败，返回失败原因 |
| 修改用户地址 | username:String  address:String | address:String | 符合预期 | 如果修改成功，返回修改后的地址；如果失败，返回失败原因 |
| 修改用户密码 | username:String  oldpassword:String  newpassword:String  confirmednewpassword:String | oldpassword:String  newpassword:String  confirmednewpassword:String | 符合预期 | 如果修改成功，返回修改后的密码；如果失败，返回失败原因 |

### 3.2.2前端

|  |  |  |
| --- | --- | --- |
| **功能** | **输入** | **测试输出** |
| 获取用户信息 |  |  |
| 修改用户姓名 |  | 566c1e23af68c04e807038008a3619a |
| 修改用户姓名 | 输入为空 | 1589985664(1) |
| 修改用户电话 |  | d241a344dd2908b42ff386c1bc57810 |
| 修改用户电话 | 输入为空 | 1589985698(1) |
| 修改用户邮箱 |  | 5f78c4134e61a57dee48943214312a4 |
| 修改用户邮箱 | 输入为空 | 1589985727(1) |
| 修改用户地址 |  | b03f760e93abc1c787658a9fd819a71 |
| 修改用户地址 | 输入为空 | 1589985754 |
| 修改用户密码 | 点击 |  |
| 修改用户密码 | 某项为空 | 1589985829(1) |
| 修改用户密码 | 密码和确认密码不同 | 1589985860(1) |

经测试，输出均符合要求

## 3.3商家用户信息管理

### 3.3.1后端

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输出** | **说明** |
| 获取用户信息 | username：String  ifpersonalUser:String | name:String phoneNumber:String email:String address:String businessLicense:String  corporateIdentity:String  category:String  collectionInformation:String | 符合预期 | 根据登录的username和用户类型返回当前商家的信息 |
| 修改商家姓名 | username:String  name:String | name:String | 符合预期 | 如果修改成功，返回修改后的姓名；如果失败，返回失败原因 |
| 修改商家电话 | username:String  phoneNumber:String | phoneNumber:String | 符合预期 | 如果修改成功，返回修改后的电话；如果失败，返回失败原因 |
| 修改商家邮箱 | username:String  email:String | email:String | 符合预期 | 如果修改成功，返回修改后的邮箱；如果失败，返回失败原因 |
| 修改商家地址 | username:String  address:String | address:String | 符合预期 | 如果修改成功，返回修改后的地址；如果失败，返回失败原因 |
| 修改商家种类 | username:String  category:String | category:String | 符合预期 | 如果修改成功，返回修改后的密码；如果失败，返回失败原因 |
| 修改商家密码 | username:String  oldpassword:String  newpassword:String  confirmednewpassword:String | oldpassword:String  newpassword:String  confirmednewpassword:String | 符合预期 | 如果修改成功，返回修改后的密码；如果失败，返回失败原因 |

### 3.3.2前端

|  |  |  |
| --- | --- | --- |
| **功能** | **输入** | **测试输出** |
| 获取用户信息 |  |  |
| 修改商家姓名 |  | b5f95261785d922379757395e0c62bb |
| 修改商家姓名 | 输入为空 | 1589985926 |
| 修改商家电话 |  | d3710b436dd1d5bcac486bef145122c |
| 修改商家电话 | 输入为空 | 1589985943(1) |
| 修改商家邮箱 |  | 589b76f875f55b93658b0b0deaa5d0f |
| 修改商家邮箱 | 输入为空 | 1589986007(1) |
| 修改商家地址 |  | c1573668ffbadd1ff24fe01ec51122f |
| 修改商家地址 | 输入为空 | 1589986023(1) |
| 修改商家种类 |  | ca835665d91d1d803892924258c06fe |
| 修改商家种类 | 输入为空 | 1589986054(1) |
| 修改商家密码 | 点击 |  |
| 修改商家密码 | 有未填项 | 1589986077(1) |
| 修改商家密码 | 密码和确认密码不同 | 1589986093(1) |

经测试，输出均符合要求

## 3.4 后台管理系统

|  |  |  |
| --- | --- | --- |
| 后台新增商家 | 填入新增商家信息 | 数据库中成功出现该商家信息 |
| 后台新增商家 | 缺少必要项 |  |
| 后台修改商家 |  | 数据库记录成功被修改 |
| 后台删除商家 |  | 该记录已成功从数据库中抹除 |
| 后台新增用户 | 缺少某项 |  |
| 后台新增用户 | 正确输入所有项 | 数据库中成功出现了新增的记录 |
| 后台修改用户 | 修改任意项 | 数据库中成功出现了修改后的记录 |
| 后台删除用户 | 删除已有记录 | 数据库中成功删除该记录 |

# 4边界测试

所有超出用户权限或不合逻辑的请求都会在业务逻辑执行前被拦截，另外，由于后端不会直接将用户传入的语句拼合成SQL语句执行，所以可以防范SQL注入攻击，而在基本安全得到保障的前提下，一些字符串内容规范性的过滤会在用户端进行。

## 4.1会话管理

|  |  |  |  |
| --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输出** |
| 用户登录 | 1. 不存在的用户名 2. 错误的密码 | 1. 提示用户不存在 2. 提示密码错误 | 与预期符合 |
| 个人用户注册 | 1. 已存在的用户名 2. 确认密码与密码不同 3. 有未填项 4. 健康状态不规范 5. 健康码不规范 6. 填入信息过长 | 1.提示用户已存在  2. 提示两次输入密码不同  3.提示有未填项  4.提示规范写法  5.提示规范写法  6.提示某项过长 | 与预期符合 |
| 商家用户注册 | 1.已存在的用户名  2.确认密码与密码不同  3.有未填项  4.填入信息过长 | 1.提示用户已存在  2.提示两次输入密码不同  3.提示有未填项  4.提示某项过长 | 与预期符合 |

## 4.2个人用户信息管理

|  |  |  |  |
| --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输出** |
| 修改用户姓名 | 1.输入为空  2.与原姓名相同 | 1. 提示输入为空 2. 提示姓名未改变 | 与预期符合 |
| 修改用户电话 | 1.输入为空  2.与原电话相同 | 1.提示输入为空  2.提示电话未改变 | 与预期符合 |
| 修改用户邮箱 | 1.输入为空  2.与原邮箱相同 | 1.提示输入为空  2.提示邮箱未改变 | 与预期符合 |
| 修改用户地址 | 1.输入为空  2.与原地址相同 | 1.提示输入为空  2.提示地址未改变 | 与预期符合 |
| 修改用户密码 | 1.输入为空  2.与原密码相同  3.原密码输入错误  4.确认密码与密码不同  5.有未填项 | 1.提示输入为空  2.提示密码未改变  3.提示原密码错误  4.提示两次输入密码不同  5.提示有未填项 | 与预期符合 |

## 4.3商家用户信息管理

|  |  |  |  |
| --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输出** |
| 修改商家姓名 | 1.输入为空  2.与原姓名相同 | 1.提示输入为空  2.提示姓名未改变 | 与预期符合 |
| 修改商家电话 | 1.输入为空  2.与原电话相同 | 1.提示输入为空  2.提示电话未改变 | 与预期符合 |
| 修改商家邮箱 | 1.输入为空  2.与原邮箱相同 | 1.提示输入为空  2.提示邮箱未改变 | 与预期符合 |
| 修改商家地址 | 1.输入为空  2.与原地址相同 | 1.提示输入为空  2.提示地址未改变 | 与预期符合 |
| 修改商家种类 | 1.输入为空  2.与原种类相同 | 1.提示输入为空  2.提示种类未改变 | 与预期符合 |
| 修改商家密码 | 1.输入为空  2.与原密码相同  3.原密码输入错误  4.确认密码与密码不同  5.有未填项 | 1.提示输入为空  2.提示密码未改变  3.提示原密码错误  4.提示两次输入密码不同  5.提示有未填项 | 与预期符合 |

## 4.4后台信息管理

|  |  |  |  |
| --- | --- | --- | --- |
| **功能** | **输入** | **预期输出** | **实际输出** |
| 修改商家姓名 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改商家电话 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改商家邮箱 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改商家地址 | 1.输入为空  2.与原地址相同 | 1.提示输入为空  2.提示地址未改变 | 与预期符合 |
| 修改商家种类 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改商家密码 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改用户姓名 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改用户电话 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改用户邮箱 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改用户地址 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |
| 修改用户密码 | 1.输入为空 | 1.提示输入为空 | 与预期符合 |

# 5持续集成

本模块开发过程中分为前端和后端两个仓库，每个仓库都提供给开发者全部权限，每个人维护自己的分支，并向Github提交代码，为了保证代码的可用性，向Github仓库添加了持续的集成支持，通过这个方法，可以减少重复过程，节省时间、费用和工作量，也可以在任何时间、任何地点生成可部署的软件，还能增强项目的可见性，从而建立团队对开发产品的信心。我们采用了这个领域运用最广的工具之一Travis CI，它对免费的开源项目可以免费使用。由于缺少统一的单元测试以及集成测试要求和代码规范，开发规约，我们不能进行集成测试，在开发过程中也没必要持续部署发布，因此只进行编译测试。这个测试对排查问题很有帮助，可以检查缺陷，了解软件的健康状况，减少假定。

# 6压力测试

压力测试是是一种基本的质量保证行为，它是每个重要软件测试工作的一部分，可以保证服务高可用性的必需品。经过压力测试后，对于服务的承载能力和可靠度才能有清晰的认识。我们目前尚未进行压力测试，决定等开发基本完成再进行。我们决定采用Apache JMeter作为测试工具。这个软件具有良好的品质保证，而且较为简单的GUI操作学习曲线平滑，不会浪费时间。同时对于我们这种前后端分离的设计，可以使用这个工具直接对服务器发起批量请求，不需要前端的参与。

# 7分析与结论

## 7.1测试情况汇总

我们的产品经历了完整的功能测试，边界测试，集成测试，本系统能够正常实现基础信息子系统的所有功能。测试基本覆盖了常见场景下可能出现的问题，但可能还存在一些未考虑到的特殊场景。总的来说已经进行的测试是开发环节中必不可少的部分，并且在测试辅助下完成了大部分开发工作，从很多地方可以看出，如果没有适当的及时的测试，很难继续完成开发工作。

## 7.2结论

我们模块的软件开发和测试缺一不可，在测试的帮助下软件开发顺利进行。但是软件的性能和服务能力等高级要求，也就是说一个可以投入应用的软件产品必须保证的指标还没有测试，没有一个实际的数据，这一点需要在以后的工作中得到实现。