

ReadTogether 测试报告

1. 概述

1.1 XX 软件概述

被测软件概况信息。

表 1 ReadTogether 软件概况

| 开发语言 | 开发环境 | 代码行数 (L) | 开发单位 | 备注 |
|--------|---|----------|----------|----|
| python | Ubuntu 12.04 Django 1.6 MySQL 5.5 | 1395 | 消除未交作业小组 | |

1.2 文档概述

本文档分为五个部分：测试报告，覆盖率报告，覆盖率分析报告，测试用例分析报告，负载测试报告，其主要内容分别如下：

测试报告：对于测试方案及内容的简述，以及对于测试效果的总体评估。

覆盖率报告：由程序自动生成，反映测试代码对于项目的覆盖率

覆盖率分析报告：根据覆盖率报告的结果进行分析，并指出未覆盖代码的比例及产生原因

测试用例分析报告：对于各测试代码的内容及功能进行说明，并将预期结果与实际结果进行比较，从对于测试效果进行分析，并了解项目目前存在的问题

负载测试报告：由程序自动生成，从性能角度对于当前项目进行测试与分析

2. 引用文档

本文档中引用到的其他文档的列表。

表 2 引用文档表

| 序号 | 标题 | 日期 | 备注 |
|----|----------|--------------|----|
| 1 | 覆盖率报告 | 2013. 12. 14 | |
| 2 | 覆盖率分析报告 | 2013. 12. 14 | |
| 3 | 测试用例分析报告 | 2013. 12. 14 | |
| 4 | 负载测试报告 | 2013. 12. 14 | |

3. 测试方案

3.1 测试级别和测试类型

本次测试的目的为检验本项目在各种可能的需求的不同样例下能否做出正确的回应

3.2 测试环境

3.2.1 软件项

测试中用到的软件

表 3 软件项

| 名称 | 用途 | 配置/版本 | 厂商/作者 | 备注 |
|--------------------|-------|--------|-------------------------------|----|
| Django unittest | 单元测试 | 1.6 | Django software foundation | |
| selenium | 集成测试 | 2.38.3 | thoughtworks | |
| coverage | 覆盖率统计 | 3.7 | Ned Batchelder and others | |
| pylot | 负载测试 | 1.26 | Corey Goldberg | |

3.2.2 硬件项

测试中用到的硬件。

表 4 硬件项

| 名称 | 用途 | 配置/版本 | 备注 |
|-------|--------|-----------------|----|
| 笔记本电脑 | 提供测试平台 | 惠普 pavilion dv6 | |

4. 测试过程简述

本次测试从开始设计到最终实际共完成耗时 11 天（12.04-12.14）。我们在本阶段采用的是将项目开发与测试设计并行进行，在确认测试的设计思路后于一到两天内将思路付诸实践的方式。

本次测试中我们共编写了 10 个测试文件，具体名称及功能如下：

test_borrow_return.py:模拟管理员完成用户借还书操作时的情景

test_info.py:检测网站的公告页面是否能正常提供信息

test_login_failure.py:监测用户登录失败时的情况

test_rank.py:查询各图书排行榜，并进入某一本书查询详细信息

test_rank_booking.py:由排行榜进入某本图书并完成预约

test_rank_update.py:实验更新图书排名时是否能正常工作

test_registers_confirm_error.py:模拟注册时认证信息错误的情况

test_registers_success.py:模拟注册时认证信息成功的情况

test_search_clickfirst_comment.py: 模拟搜索后点击第一本书, 并测试评论、纠错等功能
test_view.py:检测由于各种原因不能正常登陆网站时的情况
测试工作设计者: 武祥晋
参与执行测试的成员: 张思浩、秦同、涂珂
测试报告: 李响

5. XX 软件测试结果

5.1 测试用例执行结果

测试用例的运行结果统计。

表 7 测试用例集合

| 被测模块 | 用例总数 | 通过用例数 | 通过用例比例 |
|-------------------|------|-------|--------|
| rt/models.py | 10 | 10 | 100% |
| rt/views.py | 10 | 10 | 100% |
| rt/views_utils.py | 10 | 10 | 100% |
| rt/forms.py | 10 | 10 | 100% |

5.2 性能测试结果

详细信息见负载测试报告

6. 评估和建议

6.1 软件评估

本项目能够在所有测试用例中均正确地给出回应, 对于可能的错误境况均进行了合理的处理, 可以证明本网站正确实现了项目的要求, 并且具有较好的稳定性, 除了基本要求外, 本项目还实现评论功能和按照多种方式进行图书排名的功能, 在原有的基础上有所提高。

在性能方面, 本网站在用户极少 (小于等于 5) 时几乎所有操作均可在瞬间完成, 但在用户数量达到一定程度时 (本次测试中的极限情况是 100 名用户同时注册或搜索), 此时网站的服务虽然能够保证正确性和稳定性, 但是可以感受到一定程度的时间延迟 (搜索图书需要的时间约为 2.0s), 但考虑到设备的局限性, 我们认为这种性能速度是可以接受的

6.2 测试工作评估

本次测试工作的设计由组长武祥晋同学负责, 具体执行由张思浩、秦同、涂珂分别负责对应的工作, 本次测试在设计时的思路是针对各个模块可能出现的错误进行汇总, 然后将它们在项目实际

运行中可能导致的问题具象化，然后依此设计测试样例，并进行测试代码的编写。总体而言，本次测试对应绝大部分可能出现的问题进行了检查，得到了理想的反应，并在 test_view.py 内对部分用户可能的恶意攻击行为进行模拟，也没有出现网站被攻击成功的情况，说明网站的稳定性和安全性均令人满意

6.3 改进建议

测试过程中对于时间安排不够准确，由于前期速度没有达到预期，最后三天的负担较重，所以下一阶段一定要把工作量均匀分配

附录 1 测试用例设计

附录 2 覆盖率分析报告