



系统测试报告

---BookWise 书籍垂直搜索引擎

《软件工程管理》课程 G13 小组

组长：王伟杰

组员：安俊霖、包越、陈华杰、黄乐
诚、刘逸杰

2023.11.25

修改历史

修订日期	版本号	修改人	修改内容	审核人
2023-12-17	Version1.0	全体组员	初稿	刘逸杰
2023-12-25	Version2.0	全体组员	终稿	刘逸杰

目录

1 引言.....	4
1.1 编写目的.....	4
1.2 项目背景.....	4
1.3 定义.....	4
1.4 参考资料.....	5
1.5 测试方法.....	5
2 测试概要.....	6
2.1 测试环境.....	6
2.2 网络拓扑.....	7
3 测试计划执行情况.....	7
3.1 测试概述.....	7
3.2 测试机构和人员.....	9
3.3 测试结果.....	10
4 测试结果.....	15
4.1 Bug 趋势图.....	15
4.2 Bug 优先级分布.....	16
5 软件测试结论.....	17
5.1 功能性.....	17
5.2 易用性.....	17
5.3 可靠性.....	18
5.4 兼容性.....	18
5.5 安全性.....	18
6 测试评价.....	18
6.1 覆盖率.....	18
6.2 缺陷和限制.....	19
6.3 建议.....	19
6.4 结论.....	20
7 测试分析.....	20
7.1 测试输入搜索关键词.....	20
7.2 缺陷按轮次和级别统计.....	20
7.3 软件测试缺陷按测试类型和级别统计.....	21
7.4 软件测试缺陷按缺陷类型和轮次统计.....	22
8 典型缺陷引入原因分析.....	22

1 引言

1.1 编写目的

以下为编写此测试报告的目的：

- 对于项目整体功能进行测试
- 通过测试来对整个项目进行二次优化与调试
- 通过测试，对于项目完成度进行评价
- 为后续 bug 的修复提供方向与建议

1.2 项目背景

整个项目的目的为开发一个图书领域的垂直搜索系统，为有对于领域性书籍有获取需求的民众提供搜索结果。整个搜索引擎最终提供了关键词搜索功能，可以提供用户想看到的结果。此外，在搜索主页面，还会通过用户偏好和当前热门关键词，为用户提供其喜爱的书籍。最后，由于本项目的面向对象为普通民主，最终还提供了书籍推荐板块，帮助用户获得更多书籍的领域性知识。

总之，整个项目是一个功能覆盖面积较广，面向对象较为普遍的一个深度搜索引擎项目。

1.3 定义

普通 bug：出现以下缺陷或问题时，统称为普通 bug

- 与前端交互后，得到的结果不符合预期
- 与前端控件交互后，无法得到响应

严重 bug: 出现以下缺陷或问题时，统称为严重 bug

- 与前端控件交互后，系统卡住或崩溃
- 点击某个控件后，切换界面后，新页面显示 404 错误

1.4 参考资料

- Patton, R. (2011). Software Testing. Sams Publishing
- Naik, K., & Tripathy, P. (2008). Software Testing and Quality Assurance.

John Wiley & Sons

- G13 项目章程
- G13 项目计划
- G13 需求规格说明书

1.5 测试方法

本项目的测试工作为单元测试、集成测试与数据库测试，测试工作由测试人员单独进行完成：

1.5.1 单元测试

分为数据库单元测试与后端开发单元测试：

数据库单元测试：由数据库设计者进行测试，对于各个用到的表的增删改查操作进行检查

后端开发单元测试：完成每个模块的功能后，通过终端控制台的输出来检查每个模块的功能是否正常实现。

1.5.2 集成测试

集成测试主要通过对于前端控件使用以及交互,对于整个项目的功能点进行测试。此外,对于前端进行测试时,还可以通过 **F12** 查看前端独立的测试结果。

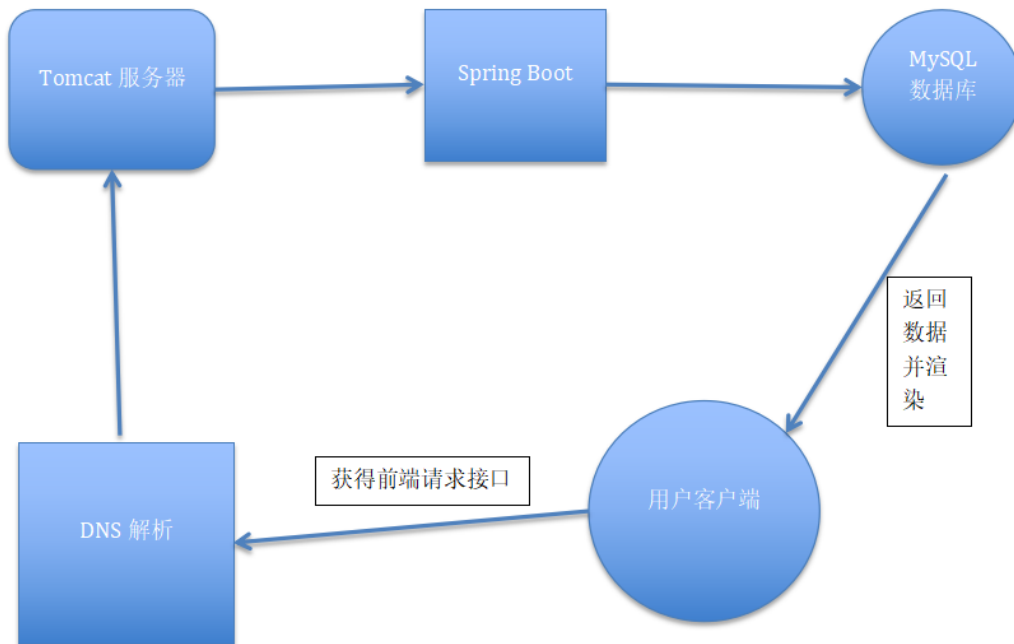
2 测试概要

整个项目的测试,从开始开发后一直在进行。开发过程中的单元测试贯穿整个开发流程;最终的集成测试时间为 2023 年 12 月 1-10 号,总共测试了四个功能点,执行 25 个测试用例。总共发现 8 个 bug,在这其中 6 个 bug 为普通 bug,即最终的测试结果与预期结果不同;剩余 2 个 bug 为严重 bug,测试后在前端展示时,产生了无法评估的结果,有的 bug 最珍贵导致了项目的运行崩溃,只能重新运行。整个书籍垂直搜索系统引擎通过了多个测试版本,除去开发时测试,最终经历一个集成的综合测试版本从而完成了最终项目的测试工作。截止至测试报告撰写完成时,所有测试版本都在预期时间内完成,而只有最终测试阶段由详细的 bug 分析表和阶段测试报告。

2.1 测试环境

硬件环境	硬件配置	软件配置	网络环境
Tomcat 服务器	PC 主机	Windows11	校网宽带
Web 服务器	PC 主机	Chrome 流量器	校网宽带

2.2 网络拓扑



3 测试计划执行情况

3.1 测试概述

测试项目名称	测试内容	测试目的
冒烟测试	对后端 Java 程序以及前后端总体运行的项目进行冒烟测试,测试一些基础功能是否能够运行	检测整个项目都各个单元是否具备最基本的功能,以便于后续测试的进行
数据库及数据库集成测试	通过 Java 文件在数据库中创建相对应的表后,通过输入 SQL 语句对需要用到的表进行增删改查	检验项目所需要用到的数据库表的设计的合法性,确保系统数据上不存在缺漏,

	操作	并能否满足项目的所需功能
用户界面测试	通过运行整个前后端构成项目后,在前端运行产生的页面中,对于用户可以进行交互的控件的功能进行测试,同时对于页面的布局,外观进行检测	保证系统的交互性,对于在于用户进行交互上仍存在的问题进行发现,并在后续进行修改
功能测试	与用户界面测试有一定重叠,根据项目计划中所设计的功能点进行测试,主要测试方法依旧未通过与前端空间的交互并对于交互结果进行检验	模拟用户实际使用系统的场景,检查系统在功能上是否仍存在较大问题
性能测试	对于系统运行时,各个功能在用户进行交互后的响应时间进行测试,同时包括在数据量较大的情况下对于响应时间进行测试	检查系统在容纳了较大的数据量时,是否能够在合理且用户可接受的范围内进行响应
负载测试	测试系统在面临大量用户同时的并发访问时,是	对于系统的并发性能进行检测,测试系统

	否会造成系统崩溃；在同时开启多个客户端访问服务器时，系统的响应性能如何	能否支持多用户的并发访问
安全测试	主要针对用户的用户密码是否能够进行安全的存储进行测试，对于同一个用户，系统是否支持多个客户端同时登录	保证基本的账号安全性，确保用户个人信息及用户使用数据的保密性，以及系统运作的稳定性

测试项目的选择和验证：

以上列出的测试项目未综合考虑整个项目运行情况的各个因素，是维护网站稳定和正常运转的必要条件。上述测试项目中的每一项对于最终项目的构建完成都十分的重要，而对于每一项测试内容，其中的具体测试方法都将在接下来进行叙述

3.2 测试机构和人员

软件管理 G13 小组中全体成员均参与开发与测试环节中。

主要包含数据库基础和编程语言基础：测试人员首先需要对 MySQL 数据的基本使用方法有一定的了解，其次，在开始测试前，需要对项目开发中用到的语言进行学习，对于项目开发中用到的框架提前进行学习，保证测试任务可以完整的进行

3.3 测试结果

3.3.1 冒烟测试

ID	描述	步骤	数据
DB_1	在数据库书籍表中插入多条数据	使用书籍数据录入 MySQL 数据库中	书籍网站
DB_2	在已录入的数据中进行查询	使用 SQL 语句多字段查询对应数据	书籍关键词，如作者、书籍类别等
FN_1	检验用户的登录功能	在 MySQL 的用户表中插入一条用户数据，在 UI 界面上登录测试	随机创造的用户名和密码数据
FN_2	检查查找书籍的基本功能	使用 UI 界面，对已经插入的书籍数据进行查找	Java 爬虫爬取的书籍信息
UI_1	检验网站的正常打开及适应性	使用 Chrome、火狐以及 Edge 浏览器分别打开网站	前后端项目后台数据

3.3.2 数据库及数据库集成测试

ID	描述	步骤	数据
DB_1	在数据库书籍表中	使用书籍数据录入	书籍网站

	插入多条数据	MySQL 数据库中	
DB_2	在已录入的数据中进行查询	使用 SQL 语句多字段查询对应数据	书籍关键词，如作者、书籍类别等
DB_3	在书籍表中增加大量书籍信息	使用 Java 程序爬取数据录入 MySQL 数据库中	Java 爬虫的来源网站
DB_4	从书籍表中删除某些缺失 ISBN	使用 SQL 语句部分删除 ISBN 为空的书籍信息	MySQL 数据库
DB_5	在数据的用户表中增加一些用户	使用 SQL 插入多条用户数据	随机构造用户信息
DB_6	在用户表使用关键词搜索用户信息	使用 SQL 的用户名信息查询指定用户	MySQL 数据库

3.3.3 用户界面测试

ID	描述	步骤	数据
UI_1	检验网站的正常打开及适应性	使用 Chrome、火狐以及 Edge 浏览器分别打开网站	前后端项目后台数据
UI_2	使用用户注册功能时是否能正确跳转	使用 UI 界面打开用户注册功能	后台 MySQL 数据库
UI_3	使用用户登录功能	使用 UI 界面，以及数据	后台 MySQL 数

	时是否能正确跳转	库测试的用户信息尝试登录	数据库
UI_4	使用书籍查询功能 是否能正确获取信息	使用关键词搜索和模块搜索, 页面是否有正确的书籍展示模块	后端 Java 程序爬取数据
UI_5	打开书籍详细展示 页面观察是否正确渲染	针对上述测试样例所展示的页面, 点击某一本书籍打开展示页面	后端 Java 程序爬取数据

3.3.4 功能测试

ID	描述	步骤	数据
FN_1	检验用户的登录功能	在 MySQL 的用户表中插入一条用户数据, 在 UI 界面上登录测试	随机创造的用户名和密码数据
FN_2	检查查找书籍的基本功能	使用 UI 界面, 对已经插入的书籍数据进行查找	Java 爬虫爬取的书籍信息
FN_3	检验用户的注册功能	在 UI 界面注册用户信息	随机创造用户数据
FN_4	检验用户偏好的书	UI 界面登录后, 观察主	Java 爬虫爬取的

	籍推荐功能是否正常运行	页面的推荐展示框中是否有相应推荐书籍	书籍信息
FN_5	使用模块搜索功能搜索书籍	在 UI 界面左侧的搜索选择中, 搜索模块化书籍数据	Java 爬虫爬取的书籍信息
FN_6	使用书籍的信息(书名、作者、语言和 ISBN 等) 能否正确精确查询书籍	在 UI 界面左侧的搜索选择中, 使用书籍固定信息搜索数据	Java 爬虫爬取的书籍信息
FN_7	针对上述搜索内容的某一书籍, 查看详细信息是否正确	在精确查询后的渲染页面上, 点击某一书籍热点, 查看其详细信息是否存在且正确	Java 爬虫爬取的书籍信息

3.3.5 性能测试

ID	描述	步骤	数据
PM_1	在用户使用书籍查询功能时, 后端爬取数据的响应时间	在前端界面使用开发者模式, 观察使用书记查询功能时, 后端返回数据的时间戳	
PM_2	用户在登录, 信息确	在 MySQL 中有大量用户	

	认的速率	信息时，检验用户登录 时所耗费的时间	
--	------	-----------------------	--

3.3.6 负载测试

ID	描述	步骤	数据
LD_1	验证系统可以为十名用户提供服务	模拟十名用户在相近时间内访问系统并进行用户登录和搜索操作，检查系统此时的相应能力	Java 程序模拟十名用户进行用户登录和搜索功能
LD_2	验证系统可以为一百名用户提供服务	模拟一百名用户在相近时间内访问系统并进行用户登录和搜索操作，检查系统此时的相应能力	Java 程序模拟一百名用户进行用户登录和搜索功能
LD_3	验证系统可以同时为一百名用户提供服务	模拟一百名用户同时访问系统并进行用户登录和搜索操作，检查系统此时的相应能力	Java 程序模拟一百名用户并行使用用户登录和搜索功能，模拟真实运行下的用户操作

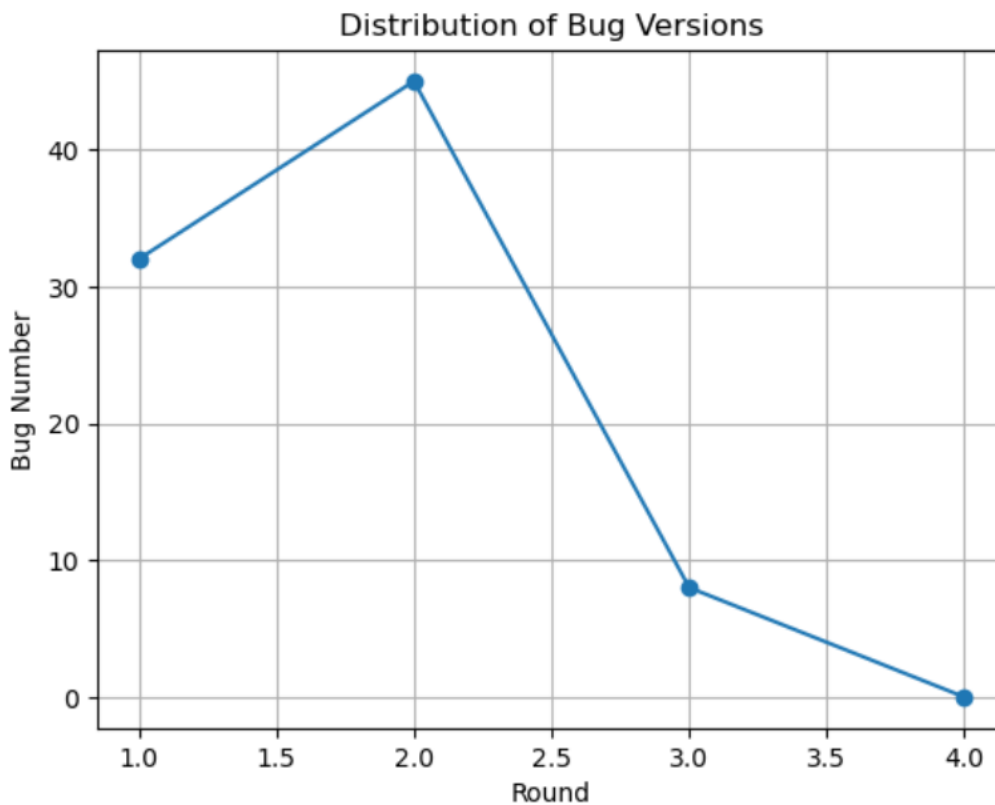
3.3.7 安全测试

ID	描述	步骤	数据
SE_1	验证系统对于未登录或注册的用户有一定的防范措施	进入系统后不进行登录，在书籍搜索页面进行功能的使用与操作	无
SE_2	边界值测试	在搜索框或用户的登录、注册框输入非法字符或空字符串，观察系统反应	无

4 测试结果

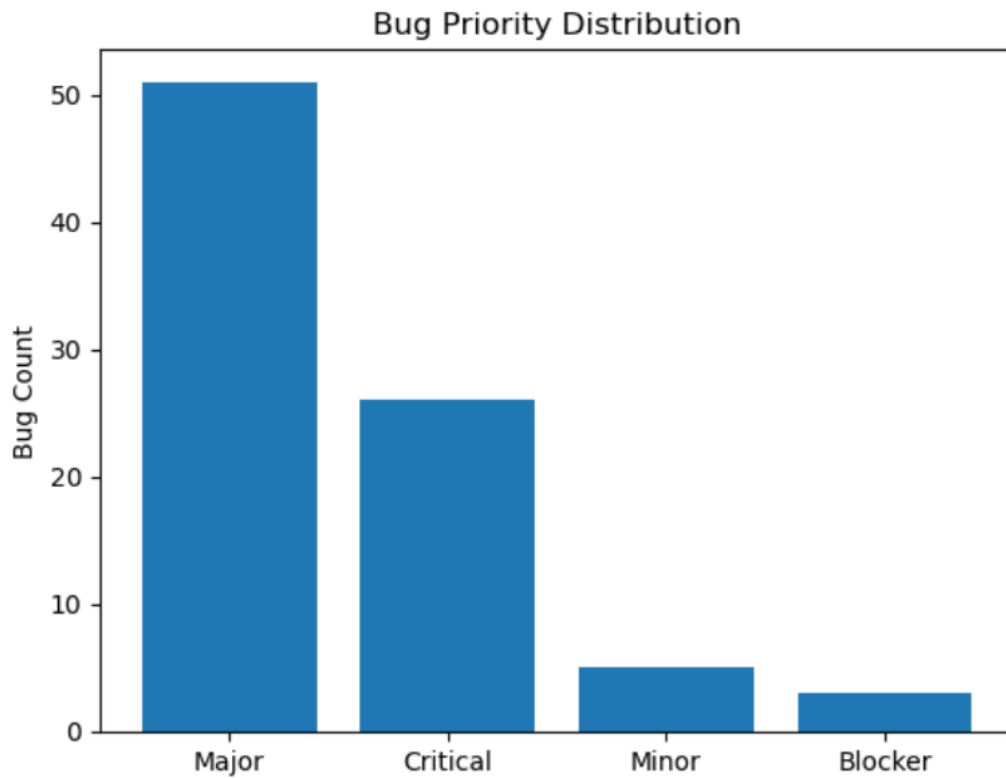
4.1 Bug 趋势图

由于集成测试仅一轮，大部分的单元测试在开发阶段已经完成，所以这里前三轮的开发测试的 bug 集是不完整的，且都已经解决无法跟踪，而第四阶段即集成测试阶段所产生的是已经解决且拥有 bug 信息的 bug List



4.2 Bug 优先级分布

出现严重级别的 bug 主要表现在以下几个方面：系统的主要功能没有实现本地数据库较大时出现程序崩溃死机；系统开发时，后台的注册信息错误导致页面无法正确跳转；后台进程在程序关闭后没有相应停止，导致程序不能启；Java 爬虫的 API 异常调用导致查询功能出错



5 软件测试结论

5.1 功能性

系统实现了搜索引擎的功能，能够根据关键词检索相关词条，获得对应领域的书籍，实现了关键词检索，书籍推荐，点赞和点评等功能

5.2 易用性

系统实现了以下易用性：

- 输入限制的正确性
- 输入限制示信息的正确性，可理解性以及一致性

系统在易用性上还存在以下缺陷：

- 输入缺少解释性说明

- 输入输出字段的可理解性差
- UI 界面排版不美观

5.3 可靠性

现有系统的可靠性控制不够严密，很多控制是通过页面控制实现的，如果页面控制失效，可以向数据库插入数据，引发错误。现有系统的容错性不高，如果系统出现严重错误，返回错误类型为找不到页面错误，则无法回复到出错前的状态

5.4 兼容性

兼容所有近两年的 Firefox 和 Chrome 以及 Safari 和 Edge 浏览器，兼容 Windows 的近三代版本，不兼容 Linux 系统和任一手机操作系统

5.5 安全性

现有系统未控制以下安全问题：

- 直接输入某一页面的 URL 能否打开页面并进行操作
- 用户注册登录所传输的信息未使用密文传输

6 测试评价

6.1 覆盖率

此次测试，所有测试用例都是在中文界面下执行，未在英文界面下执行，测试不包括英文界面下的测试，也不包括英文翻译的测试。此次

测试，部分页面需求无明确的定义，对输入限制无详细定义或描述，无明确的测试依据。在测试过程中，测试是根据输入字段含义，加之测试人员理解，以及和项目经理，开发人员沟通获得测试依据，无法保证测试依据的正确性和完整性。因此，没有进行完整的，正确的无效数据的测试，测试覆盖率不够，无法保证测试的有效性和正确性。

6.2 缺陷和限制

由于软件架构并不复杂，仅使用前端 Vue+后端 Spring boot 的架构，所以在实现和完成阶段进行测试中，没有暴露出过多问题，基本有出现的大问题或者可解决的小问题都在开发阶段解决了，并不会出现网站无法跳转，页面卡死等情况。但对于软件层面而言，无法满足多操作系统的环境，如无法应用与 Linux 系统和手机系统是关键限制。同时，因为拥有用户功能，没有采取密文传输也是安全方面的欠妥考虑。

6.3 建议

在项目开始时，应该制定编码标准、传输密文格式以及数据库范式标准，需求变更更准确，开发和测试人员都严格按照标准进行，可以在后期减少因为开发，测试不一致而导致的问题，同时也可以降低沟通成本。发布版本的时候，正确布置测试环境，减少因为测试环境，测试数据库数据的问题而出现的无效 bug。开发人员解决 bug 的时候，填写 bug 原因以及解决方式，方便 bug 的跟踪。开发人员在开发版本上发现 bug，可以通知测试人员，因为开发人员发现的 bug 仍有可能

在测试版本上出现，而测试人员和开发人员的思路不同，有可能测试人员没有发现的开发人员提供的 bug。而且，这样可以保证发现的 bug 都能够被跟踪，且当各个模块拼接时，进行接口的数据类型和顺序匹配会更加顺畅。

6.4 结论

本系统已经满足甲方出的基本需求以及乙方最开始的项目策划书，所以此系统的基本能力已经完备，可以实现交付。

7 测试分析

7.1 测试输入搜索关键词

测试输入的关键词

序号	输入关键词
1	正常关键词
2	屏蔽词
3	无法识别关键词
4	空字符

7.2 缺陷按轮次和级别统计

	首轮测试	第一轮回归 测试	集成测试	总计

致命缺陷	1	1	0	2
严重缺陷	2	1	2	5
一般缺陷	5	3	3	11
建议缺陷	8	5	1	14
小计	16	10	6	32

7.3 软件测试缺陷按测试类型和级别统计

	首轮测试				第一轮回归测试				集成测试			
测试结果	致命	严重	一般	建议	致命	严重	一般	建议	致命	严重	一般	建议
文档审查	2	1	1	0	0	2	1	1	0	0	2	2
功能测试	1	4	2	3	0	3	3	1	0	2	1	1
性能测试	0	0	2	1	0	0	0	0	0	0	0	0
边界测试	0	0	0	1	0	0	2	0	0	0	0	2
接口测试	1	0	0	0	0	0	2	0	0	0	0	0
交互测试	0	0	2	0	0	0	0	0	0	0	0	0

试												
安全性	0	0	1	0	0	0	0	1	0	0	0	0
总计	4	5	8	5	0	5	8	3	0	2	3	5

7.4 软件测试缺陷按缺陷类型和轮次统计

	首轮测试	第一轮回归 测试	集成测试	总计
程序缺陷	4	2	2	8
文档缺陷	2	2	0	4
设计缺陷	8	1	1	10
其他缺陷	1	1	0	2
小计	15	6	3	24

8 典型缺陷引入原因分析

测试过程中发现的缺陷主要有以下几个方面：

- 需求定义不明确

需求文档中，存在功能定义错误，输入输出字段描述错误，输入输出字段限制定义错误，输入输出限制定义缺失这几种类型的缺陷，使得开发人员根据需求进行设计时，没有考虑相关功能的关联性，以及需求错误的地方。在测试过程中，需求相关的问题表现出来，想要做更

正，同时设计也必须跟着改动，浪费时间和影响开发人员的积极性，降低开发人对需求的信任，看你会导致开发人员不按照需求进行设计而根据自己的经验来进行设计

- 功能性错误

功能没有实现，导致无法进行需求规定的功能的测试，主要是无法进入用户登录层面或者书籍详细信息查询页面，还有部分书籍信息错误等功能问题。功能实现错误，实现了需求未定义或未实现需求定义功能，甚至执行需求定义的功能时系统错误，主要是查询时，多重关键词功能的缺失。

- 页面设计和需求不一致

页面设计没有根据需求进行，输入输出字段文字错误，用户无法理解字段含义。页面设计没有完成需求规定的输入限制验证，如密码和用户名限制，导致用户可以输入错误的或者无效的数据吗，这些数据有可能会引起功能性错误。

- 页面设计易用性缺陷

页面设计不友好，系统中很多页面的输入字段无明确的输入示例，用户无法理解何种输入是正确的，但是用户输入错误后，系统显示出错，增加用户负担。示例信息错误，不同模块相同结果的示例信息不一致，用户操作后，相应的示例信息不明确，引起用户误解。