
2013 年全国大学生信息安全竞赛作品测试报告

作品名称： 基于云计算的微博敏感信息挖掘系统

组 长： 吕若尘

组 员： 唐瞻立，李声龙，史乔茜

提交日期： 2013-7-25

目录

1. 引言.....	3
1.1. 编写目的	3
1.2. 开发单位	3
2. 评价标准.....	3
2.1. 范围	3
2.2. 数据整理	3
2.3. 尺度	4
3. 计划.....	4
3.1. 软件说明	4
3.2. 测试目的和要求	6
3.2.1. 测试目的.....	6
3.2.2. 测试要求.....	7
3.2.3. 团队责任.....	7
3.2.4. 测试方法.....	8
3.2.5. 测试策略.....	9
3.2.6. 测试任务与进度.....	9
3.2.7. 风险管理.....	10
3.2.8. 测试具体内容.....	11
3.3. 测试环境	13
3.3.1. 硬件环境.....	13
3.3.2. 软件环境.....	14
3.4. 测试案例设计	14
3.4.1. 客户端功能测试.....	14
3.4.2. 服务端功能测试.....	31

1. 引言

1.1. 编写目的

编写测试报告的目的是通过整个测试过程，包括测试计划、测试设计、案例编写、测试总结等步骤去了解基于云计算的微博敏感信息挖掘系统的质量如何。是否存在缺陷，若存在缺陷则其原因是什么以及该如何修复。希望在本系统发布之前能将缺陷修复。同时向用户呈现本系统的具体测试细节，使用户能够明白系统的软件缺陷以及具体的操作方法，这将便于用户理解和使用本系统。本文档面向的读者主要是项目的开发人员和测试人员。

1.2. 开发单位

Sphinx 团队，团队由以下成员组成：

吕若尘（队长）、唐瞻立、李声龙、史乔茜。

2. 评价标准

2.1. 范围

本测试案例采用动态黑盒测试¹的方式，根据现有的需求规格说明书进行测试案例的设计和选择，仅适用于对系统的正确性测试。测试案例将针对系统的核心功能进行测试，主要包括用户影响力计算，敏感用户挖掘与追踪以及敏感话题抓取和热门微博话题获取等。

2.2. 数据整理

为了把测试数据加工成便于评价的适当形式，使得测试结果可以同已知结果进行比较。本测试采用手工的方式进行数据的整理，撰写测试报告和统计结果，

¹ 不深入代码细节的软件测试方法称为动态黑盒子测试。

便于测试人员与程序开发人员之间的交流，也使客户更加直观的理解和查看软件的测试结果。

2.3. 尺度

- 1. 系统正确的完成了所有需求规格说明书规定的功能；
- 2. 测试输出结果与预期输出结果之间仅允许有细微的界面偏差，输入、输出的数据必须准确。

3. 计划

3.1. 软件说明

功能编号	功能名称	输入	输出
客户端功能			
TSC-01-01	用户登录	用户名，密码	成功进入系统主页
TSC-01-02	查看敏感用户列表	1. 进入系统主页 2. 点击敏感用户列表	跳转到敏感用户跟踪页面，显示敏感用户列表
TSC-01-03	对敏感用户按照微博影响力降序进行查看	1. 进入敏感用户跟踪页 2. 在敏感用户列表上点击PageRank 值列表头	敏感用户列表按照其微博用户影响力进行从高到低排序
TSC-01-04	对敏感用户按照敏感值进行查看	1. 进入敏感用户跟踪页 2. 在敏感用户列表上点击敏感值列表头	敏感用户列表按照其微博用户敏感值进行从高到低排序

TSC-01-05	按照关键词对敏感用户进行过滤	<ol style="list-style-type: none"> 1. 进入敏感用户跟踪页 2. 在敏感用户过滤搜索栏填入敏感词 	敏感用户列表中筛选出相关敏感词的敏感用户
TSC-01-06	查看敏感用户的详细信息和微博滚动列表	<ol style="list-style-type: none"> 1. 进入敏感用户跟踪页 2. 在敏感用户列表上点击任意用户名 	跳转到敏感用户信息页面，显示其个人信息和敏感微博
TSC-01-07	添加敏感事件话题语料库关键词	<ol style="list-style-type: none"> 1. 进入系统首页 2. 点击系统设置按钮 3. 点击自定义敏感语料库栏，输入敏感词添加 	敏感词添加成功
TSC-01-08	查看话语权威列表	<ol style="list-style-type: none"> 1. 进入系统首页 2. 点击话语权威列表 	跳转到微博话语权威页面，显示话语权威列表
TSC-01-10	查看敏感话题关键词	<ol style="list-style-type: none"> 1. 进入系统首页 2. 点击敏感话题关键词 	跳转到敏感话题关键词页面，显示敏感话题关键词图表
TSC-01-11	查看热门微博话题	<ol style="list-style-type: none"> 1. 进入系统首页 2. 点击热门微博话题 	跳转到近期热门微博话题页面，显示近期微博话题图表
TSC-01-12	查看敏感用户地理分布	<ol style="list-style-type: none"> 1. 进入系统首页 2. 进入敏感用户页面 3. 敏感用户地理位置分布图 	跳转到敏感用户地理位置分布页面，显示地理位置分布图

TSC-01-13	查看敏感用户社交关系网	1. 进入系统首页 2. 进入敏感页面 3. 敏感用户社交关系网	跳转到敏感用户社交关系页面，显示敏感用户社交关系网
服务器功能			
TSS-02-01	添加用户	1. 开启服务 2. 进入用户管理终端 3. 输入添加用户命令，输入账号密码	用户添加被添加
TSS-02-02	删除用户	1、开启服务 2. 进入用户管理终端 3. 输入删除用户命令，输入用户账号	用户被删除

表 1-1 软件测试计划说明

3.2. 测试目的和要求

3.2.1. 测试目的

1. 在产品达到预期目的，是否实现项目设计要求；
2. 产品可以稳定运行，并易于操作；
3. 系统产品 Bug 在允许范围之内；
4. 缺陷率在可控制范围内。

3.2.2. 测试要求

1. 以 80/20 原理² 为指导，做到在有限的时间里发现尽可能多的缺陷，尤其是严重缺陷；
2. 通过测试，不断完善测试用例；
3. 根据事先定义的测试执行顺序进行测试，并填写测试记录表，保证测试过程是受控制的；
4. 测试重点放在各子系统的功能实现上，问题较多的是按键选择性问题。

3.2.3. 团队责任

任务	李声龙	吕若尘	唐瞻立	史乔茜
制定统一的测试方案	✓			
制作测试总体进度	✓			
制定测试的总体设计	✓			
编写测试案例说明		✓	✓	✓
客户端测试		✓		
服务器端测试				✓
动态黑盒测试	✓	✓	✓	✓

² 80% 的软件缺陷常常生存在软件 20% 的空间里。

缺陷报告跟踪与统计		✓	✓	
回归测试		✓	✓	

表 1-2 项目测试团队责任

3.2.4. 测试方法

3.2.4.1. 动态黑盒测试

测试要求	描述
测试目标	该项目执行过程中的实际行为能力
测试范围	功能性测试和数据性测试
技术	1. 确认软件至少能做什么,即通过性测试 2. 通过各种手段进行失效性测试
开始标准	项目正常运行
完成标准	项目功能正常显示。
测试重点和优先级	1. 通过性测试 2. 失效性测试 3. 数据测试
需考虑的特殊事项	无

表 1-3 动态黑盒测试计划表

3.2.5. 测试策略

- 1. 测试计划与需求制定、用例设计同步进行；
- 2. 必须制定测试需求：通过确定要测试的内容和各自的优先级、重要性，使测试设计工作更有目的性，在需求的指导下设计出更多更有效的用例；
- 3. 逐步完善测试用例库：测试用例库的建设是一个不断完善的过程，我们要在有限的时间里，先设计出一整套的测试用例，重要的部分用例需要设计得完善一些，一般部分的则指出测试的要点，在以后的测试工作中再不断去完善测试用例库；
- 4. 测试过程要受到控制：根据事先定义的测试执行顺序进行测试，并填写测试记录表，保证测试过程是受控的；
- 5. 采用人工测试对软件功能进行动态黑盒测试。

3.2.6. 测试任务与进度

测试活动	计划开始日期	实际开始日期	结束日期
制定测试计划	2013/04/10	2013/04/10	2013/04/12
制定测试方案	2013/04/13	2013/04/14	2013/04/22
功能测试	2013/04/22	2013/04/22	2013/05/18
测试结果评估	2013/05/20	2013/05/24	2013/05/26
测试缺陷追踪	2013/05/26	2013/05/26	2013/05/27
回归测试	2013/06/01	2013/06/04	2013/06/04
项目提交	2013/06/10	2013/06/10	2013/06/10

表 1-4 测试任务与进度

3.2.7. 风险管理

序号	风险描述	严重度	缓解措施
1	系统不稳定或卡死	9	找出系统不稳定原因，以及耗损内存模块，进行代码优化。
2	功能不完善，没有完成需求规定功能	9	确定没有完善的功能，然后进行功能代码编写。
3	模块集成出错导致功能无法实现，或功能出现紊乱	8	针对某一模块进行修改，加工与测试，然后对几个模块进行集成测试，确认功能。
4	修改某模块时导致其它功能不能使用	8	代码检查，一般逻辑错误，或代码还原，恢复修改前状态。
5	核心技术无法攻破，导致项目滞后	7	选用可实现的技术，降低风险，请教指导老师。
6	软件核心开发人员有事调离团队	7	团队任务分派时，指定两个或两个以上成员担任开发人员。

7	在项目后期中发生软件需求变更	9	项目设计时，应使项目的架构具有良好可扩展性和模块性。
8	系统出现一些小bug	5	对于不影响系统正常运行的bug可以不处理，锁定bug的原因，对相应代码进行优化。
9	界面有小瑕疵	3	在时间允许的情况下，可以进行界面优化。

表 1-5 风险管理

3.2.8. 测试具体内容

案例编号	案例名称	测试目的	进度安排
	用户成功登陆	验证输入正确的用户名和密码，是否成功跳转系统主页	2013-04-22
	用户登录	验证系统是否可以提示错误信息	2013-04-22
	用户按照微博影响力对敏感用户进行降序查看	验证敏感用户的微博影响力是否被正确计算	2013-05-05

TC-01-04	用户输入关键词对敏感用户进行过滤	验证系统能否通过关键词对敏感用户进行过滤	2013-04-28
TC-01-01	用户查看敏感用户的资料和微博	验证系统能否对敏感用户进行追踪	2013-05-08
TC-01-02	用户手动添加敏感事件关键词	验证用户能否手动向语料库添加	2013-05-08
TC-01-03	用户查看话语权威列表	验证系统能否正确计算微博用户影响力并进行排序	2013-05-13
TC-01-08	用户查看敏感用户实时微博	验证系统能否显示敏感用户实时微博	2013-05-13
TC-01-09	用户查看微博热门话题与敏感词	验证系统能否抓取敏感话题关键词	2013-05-13
TC-01-10	导出系统数据	验证系统能否导出	2013-05-15
TC-01-11	查看敏感用户地理位置分布图	验证系统能否生成敏感用户的地理位置分布图	2013-05-15

TC-01-12	查看敏感用户社交关系图	验证系统能否生成敏感用户的社交关系图	2012-05-15
TC-02-01	管理员添加用户	验证管理员能否添加用户	2013-04-30
TC-02-02	管理员删除用户	验证管理员能否删除用户	2013-04-30

表 1-6 测试内容表

3.3. 测试环境

3.3.1. 硬件环境

使用三台电脑（HW-01、HW-02、HW-03）利用 MongoDB 分布式功能，搭建私有云平台。

HW-01/ HW-02/ HW-03	环境参数
硬件设备	联想扬天 T4980d
处理器	酷睿三代 i5 处理器 I5 3470
内存	4GB DDR3
硬盘	1TB SATA 串行 7200 转/分钟
网络适配器	千兆网卡

表 1-7 硬件环境表

3.3.2. 软件环境

SW-01	环境参数
操作系统	Ubuntu 12.04 LTS
Ruby 环境	Ruby 1.9.3
Python 环境	Python 2.7
Web 浏览器	Chrome 26.0.1410.63
服务器 IP 地址及端口号	localhost:3000

表 1-8 软件环境表

3.4. 测试案例设计

3.4.1. 客户端功能测试

3.4.1.1. 客户端功能测试案例一

测试用例 ID	DT-01-01
测试用例名称	用户成功登陆
测试用例说明	验证用户成功登录系统
预置条件	服务器处于打开状态
输入	1. 打开浏览器 2. 输入 localhost:3000，连接到服务器，显示登录界面 3. 输入正确的用户名和密码
预期结果	用户验证成功，进入系统主页

实际结果

实际结果与预期结果一致

A login form on a teal background. It has a 'Username:' label above a white input field containing the placeholder text 'Enter your name'. Below that is a 'Password:' label above another white input field containing the placeholder text 'Password'. To the right of the password field is a blue link that says 'Lost Password?'. At the bottom center is a red button with the text 'Sign in' in white.

图 1-1 系统登录界面

A screenshot of the Sphinx web analytics dashboard. The top navigation bar includes the 'Sphinx' logo, a 'Toggle menu' button, a 'Username' dropdown menu, and a 'Log out' button. The main content area is titled '微博用户' (Weibo Users) and features a table of '敏感用户列表' (Sensitive User List). The table has columns for user names, associated keywords, PR scores, and counts. To the right of the table are sorting options: '敏感度排序' (Sort by Sensitivity), '影响力排序' (Sort by Influence), and '自定义排序' (Custom Sort). Below the table is a '话题关键词' (Topic Keywords) section. It includes a '微博热门话题' (Weibo Hot Topics) list with 7 items, a bar chart showing '微博 24 小时热门话题' (Weibo 24-hour Hot Topics) with a comparison to yesterday, and a '趋势' (Trend) line chart showing '话题关键词趋势' (Topic Keyword Trends) over time. The bottom of the dashboard has a footer with the text 'Designed and built with all the love in the world by @kshift. Copyright © 2013 Sphinx. All Rights Reserved.'

图 1-2 进入系统主页

结果分析

测试通过

表 1-9 客户端功能测试案例一

3.4.1.2. 客户端功能测试案例二

测试用例 ID	DT-01-02
测试用例名称	用户非法登录
测试用例说明	验证用户非法登录系统
前置条件	服务器处于打开状态
输入	1. 打开浏览器 2. 输入 localhost:3000，连接到服务器，显示登录界面 3. 输入错误的用户名和密码
预期结果	用户验证失败，页面提示用户名或密码错误
实际结果	<div>实际结果与预期结果一致</div> <div></div> <div>图 1-3 用户验证失败界面</div>
结果分析	测试通过

表 1-10 客户端功能测试案例二

3.4.1.3. 客户端功能测试案例三

测试用例 ID		DT-01-03	
测试用例名称		按微博用户影响力降序查看敏感用户	
测试用例说明		验证敏感用户列表能否按照用户微博影响力降序显示	
预置条件		1. 服务器处于打开状态 2. 进入敏感用户列表页面	
输入		在敏感用户列表上点击 PageRank 值表头	
预期结果		敏感用户按照微博用户影响力进行降序排列	
实际结果		实际结果与预期结果一致	

敏感用户列表	1-5 of 50	<	>
假装在纽约	中纪委、三聚氰胺、校长、海南万宁校长、丁锦昊	PR 6.2	29
成都冯川	丁锦昊、四川渠县档案局、无德无信中国人、埃及旅游、金正恩	PR 4.3	67
记者张颖	湖南唐慧、渔民、儿童色情、山西矿难、联名抗议	PR 5.6	58
溜溜之旁观	金正恩、崔龙海、丁锦昊、海南万宁、三聚氰胺	PR 4.5	37
闻宾Jerry	刷肉育骨、渔民、不能自由呼吸、戴手铐游街示众、中纪委	PR 3.9	48

图 1-4 敏感用户列表页面

结果分析	测试通过
------	------

表 1-11 客户端功能测试案例三

3.4.1.4. 客户端功能测试案例四

测试用例 ID	DT-01-04
测试用例名称	按照关键词对敏感用户进行过滤
测试用例说明	验证系统能否按照关键词对敏感用户进行过滤
预置条件	1. 服务器处于打开状态 2. 进入敏感用户列表页面
输入	在敏感用户过滤搜索栏上输入关键词
预期结果	敏感用户列表显示按照关键词过滤出的敏感用户
实际结果	实际结果与预期结果一致



图 1-5 敏感用户列表页面

结果分析

测试通过

表 1-12 客户端功能测试案例四

3.4.1.5. 客户端功能测试案例五

测试用例 ID	DT-01-05
测试用例名称	查看敏感用户详细信息和微博滚动列表
测试用例说明	验证系统追踪敏感用户
预置条件	1. 服务器处于打开状态 2. 进入敏感用户列表页面
输入	在敏感用户列表中点击任意敏感用户
预期结果	进入敏感用户信息页面，显示用户详细信息和敏感微博
实际结果	实际结果与预期结果一致



3.4.1.6. 客户端功能测试案例六

测试用例 ID	DT-01-06
测试用例名称	手动添加敏感事件话题关键词
测试用例说明	验证系统能否手动添加敏感事件话题关键词
预置条件	1. 服务器处于打开状态 2. 进入敏感用户列表页面
输入	1. 点击系统设置按钮 2. 点击自定义敏感事件语料库栏，输入敏感词
预期结果	敏感词添加成功
实际结果	实际结果与预期结果一致



图 1-7 设置页面

表 1-14 客户端功能测试案例六

3.4.1.7. 客户端功能测试案例七

测试用例 ID	DT-01-07
测试用例名称	查看话语权威列表
测试用例说明	验证系统能否显示话语权威列表
预置条件	1. 服务器处于打开状态 2. 进入敏感用户列表页面
输入	点击微博话语权威列表
预期结果	进入微博话语权威页面，显示微博话语权威
实际结果	<div>实际结果与预期结果一致</div> <div><div>话语权威用户列表</div><div><div>1. 假装在纽约 PR 4.3</div><div>纽约市人民政府驻北京办事处主任 著名话剧演员 联络：mrjiazhuang@gmail.com</div><div>2. 记者张颖 PR 5.6</div><div>今日说法上会记者</div><div>3. 成都冯川 PR 6.2</div><div>四川大学中文系退休教授、锦城学院外语系在职主任。治不中不西之学，说有对有错的话。</div><div>4. 潘潘之旁观 PR 4.5</div><div>一点一滴地改过，一点一滴地向善。</div><div>5. 闻宾Jerry PR 3.9</div><div>双子座AB男！关注移动互联网，开放平台，社交应用。</div></div><div><div>Prev</div><div>1</div><div>2</div><div>3</div><div>4</div><div>5</div><div>Next</div></div></div> <div>图 1-8 微博话语权威页面</div>
结果分析	测试通过

表 1-15 客户端功能测试案例七

3.4.1.8. 客户端功能测试案例八

测试用例 ID	DT-01-8
测试用例名称	查看敏感用户实时微博
测试用例说明	验证系统能否显示敏感用户实时微博
预置条件	1. 服务器处于打开状态 2. 进入敏感用户实时微博页面
输入	点击敏感用户实时微博菜单
预期结果	进入敏感用户实时微博页面，显示敏感用户实时微博
实际结果	实际结果与预期结果一致

Sphinx

Toggle menu

Username

Log out

暂停滚动



成都冯川

Tue Jun 04 21:14:08 2013

退休前在川大上西方文化课，因时间关系，最多讲到19世纪上半叶。其实下半叶一直到一战这一时段很值得关注。此时，普通民众处境悲惨；从基督教人道主义中发展出社会主义；甚至古典自由主义也转而主张以政府干预改善教育和民生；而信奉社会达尔文主义丛林法则的掌权者却沉醉于走向帝国强盛的光荣梦。



成都冯川

Tue Jun 04 20:24:50 2013

1844年6月4日，世界上最后两只大海雁（garefowl）在冰岛西南海岸之外的岛上被人类无情屠杀。此物种从此绝种。大海雁是一种企鹅式的翅膀很小不能飞的鸟类，它的标本现存欧洲一些博物馆里。



成都冯川

Tue Jun 04 20:07:09 2013

1913年6月4日，英国积极争取女性选举权的艾米丽·戴维孙（Emily Davison）突然冲进赛马场，投身于英王乔治五世正在奔跑的爱马蹄下并于四天后死去。她生前曾多次坐牢绝食，死后，英国妇女社会政治联盟（Women's Social and Political Union）为她举行了隆重葬礼。



假装在纽约

Tue Jun 04 13:12:00 2013

有没有哪些明星好像很厉害很受人喜欢可是你就是喜欢不起来的？我的答案是包括但不限于朱茵，郭晶晶，刘若英。



人人都是自己的历史学家

Tue Jun 04 07:23:36 2013

当然，谁也不知道明天/明天从另一个早晨开始/那时我们将沉沉睡去



假装在纽约

Mon Jun 03 23:10:02 2013

敢住重庆大厦的，不是太无知就是太莽撞。香港人都知道那是整个香港最乱的地方。我进去转过一圈，拍了张店铺的照片，被黑人店主凶神恶煞夺过相机删掉。再穷游也要以安全为前提。如果是因为香港酒店贵，不如住深圳，每天罗湖过关到尖沙咀也不过一个小时。如果是因为王家卫的电影，只能说文艺青年没脑子。



成都冯川

Sun Jun 02 18:19:14 2013

#川大往事# 川大中文系教授令人高山仰止，不仅因学问，更因其风骨。当时我们宿舍都很爱冯川，尤其是 @雕鸮爱小宝 同学更是如此。（此条来自网友 @观察西岸）



人人都是自己的历史学家

Sun Jun 02 09:41:36 2013

亚翁尝言，一个脱离国家而能生活的人非兽即神。我看到大群野兽将如神灵降临此国，如乌云般在大街上奔腾。

图 1-9 敏感用户实时微博页面

结果分析

测试通过

表 1-16 客户端功能测试案例八

3.4.1.9. 客户端功能测试案例九

测试用例 ID		DT-01-9
测试用例名称	查看近期微博热门话题与敏感词	
测试用例说明	验证系统能否获取近期微博热门话题与敏感词	

前置条件	1. 服务器处于打开状态 2. 进入热门话题与敏感词列表页面
输入	点击敏感话题关键词菜单
预期结果	进入热门话题与敏感词列表页面，显示敏感话题关键词图表
实际结果	<p>实际结果与预期结果一致</p> <div><p>敏感事件关键词</p><p>本周被提到最多的关键词如下： (右图为本周与上周对比)</p><ol style="list-style-type: none">1. 不能自由呼吸2. 海南万宁校长3. 丁锦昊4. 645. 新闻自由6. 山西矿难7. 彭州PX</div> <p>图 1-10 敏感话题关键词</p> <div><p>微博热门话题</p><p>微博 24 小时热门话题，由新浪官方提供： (左图为今日与昨日对比)</p><ol style="list-style-type: none">1. 随手拍2. 周五的微心情3. 禁播大漠谣4. 524陈制作漂流起航5. 半透明校服6. 最想去旅行的地方7. 小米手机就是快</div> <p>图 1-11 微博热门话题</p>
结果分析	测试通过

表 1-17 客户端功能测试案例九

3.4.1.10. 客户端功能测试案例十

测试用例 ID	DT-01-10
测试用例名称	导出系统数据
测试用例说明	验证系统能否导出数据
预置条件	1. 服务器处于打开状态 2. 进入数据输出页面
输入	点击数据输出菜单
预期结果	进入数据输出页面，显示数据输出选项
实际结果	实际结果与预期结果一致

SphinxToggle menuUsernameLog out

选择输出信息

敏感用户列表ON

敏感用户滚动微博OFF

敏感事件关键词ON

话语权威用户列表OFF

微博热门话题ON

选择输出格式

☒ PDF

☐ HTML

☐ PNG

导出数据

Designed and built with all the love in the world by @kshift.
Copyright © 2013 Sphinx. All Rights Reserved.

图 1-12 数据输出页面

结果分析	测试通过
------	------

表 1-18 客户端功能测试案例九

3.4.1.11. 客户端功能测试案例十一

测试用例 ID	DT-01-11
测试用例名称	查看敏感用户地理分布
测试用例说明	验证系统能生成敏感用户的地理分布图
预置条件	1. 服务器处于打开状态 2. 进入敏感用户页面
输入	点击查看敏感用户地理位置分布
预期结果	进入用户地理位置分布页面，显示用户地理位置分布图
实际结果	实际结果与预期结果一致



图 1-13 敏感用户地理位置分布页面

结果分析	测试通过
------	------

表 1-19 客户端功能测试案例十一

3.4.1.12. 客户端功能测试案例十二

测试用例 ID	DT-01-12
测试用例名称	查看敏感用户社交网图
测试用例说明	验证系统能生成敏感用户的社交网图
预置条件	1. 服务器处于打开状态 2. 进入敏感用户页面
输入	点击查看敏感用户社交网图
预期结果	进入敏感用户社交网页面，显示敏感用户社交网图
实际结果	实际结果与预期结果一致

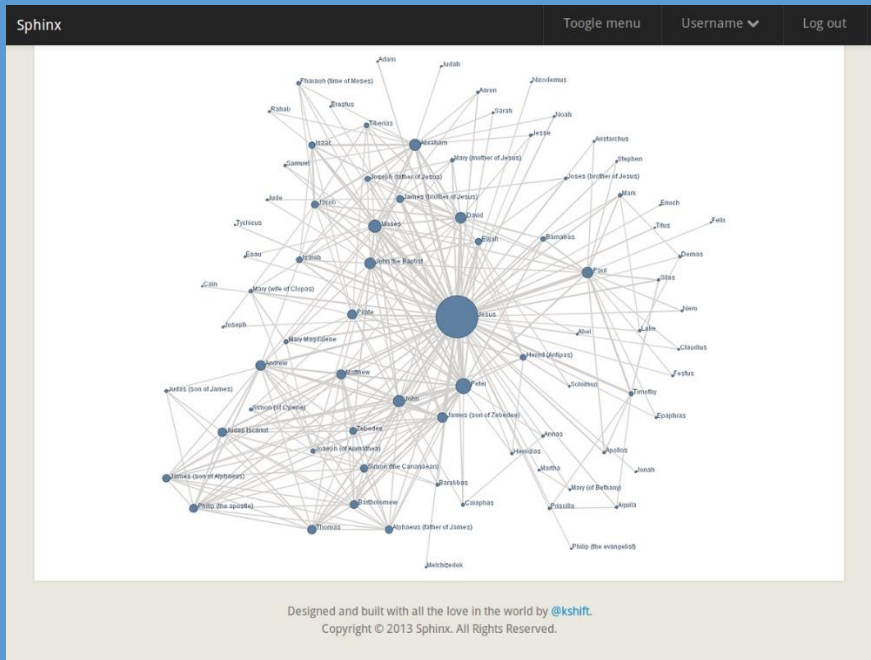


图 1-14 敏感用户社交关系网页面

结果分析	测试通过
------	------

表 1-20 客户端功能测试案例十二

3.4.2. 服务端功能测试

3.4.2.1. 服务功能测试案例一

测试用例 ID	DT-02-01
测试用例名称	添加系统用户
测试用例说明	验证能否添加系统用户是否获取近期微博热门话题
预置条件	服务器处于打开状态
输入	1. 打开服务终端 2. 输入添加用户命令 3. 输入新用户的用户名和密码
预期结果	终端提示添加用户成功
实际结果	实际结果与预期结果一致 
结果分析	测试通过

图 1-15 服务终端

表 1-21 服务功能测试案例一

3.4.2.2. 服务功能测试案例二

测试用例 ID	DT-02-02
测试用例名称	删除系统用户
测试用例说明	验证能否删除系统用户是否获取近期微博热门话题
预置条件	服务器处于打开状态
输入	1. 打开服务终端 2. 输入添加用户命令 3. 输入新用户的用户名和密码
预期结果	终端提示删除用户成功
实际结果	实际结果与预期结果一致  <p>The screenshot shows a terminal window titled 'rails console'. The user enters the command 'rails console' and the environment is loaded. Then, they enter 'u = User.find_by_name("Oedipus")'. The console shows the user object details. Next, they enter 'u.destroy'. The console shows a transaction being started, a DELETE SQL statement being executed, and the transaction being committed. Finally, they enter 'User.all' and the console shows an empty array '[]', indicating the user has been successfully deleted.</p>
结果分析	测试通过

表 1-22 服务功能测试案例二