**测试方案**

1. **测试计划**

1.1概述

本次小学期课程中，测试组承担智能起草演讲稿系统的测试工作。开发一个适用于用户编写演讲稿的网站，实现网上智能起草和编辑演讲稿。 主要实现功能：

智能起草功能：用户输入演讲稿关键要素，自动生成演讲稿模板。

编辑功能：用户根据自己的需要编辑生成的演讲稿模板。

本次测试的主要目的是确保软件系统功能符合需求，且可以为用户提 供良好的用户体验。

1.2测试分工以及进度控制

|  |  |  |
| --- | --- | --- |
| 测试模块 | 负责人 | 备注 |
| 需求文档测试 | 李家葶 |  |
| 设计文档测试 | 王连宁 |  |
| 代码分析 | 范扬帆，王连宁 |  |
| 界面分析 | 陈有贤 |  |
| 功能测试 | 陈有贤 |  |
| 性能测试 | 敬甲男 |  |
| 安全性测试 | 彭堂智 |  |
| 易用性测试 | 陈有贤 |  |

4.4测试资源与环境

4.4.1 人力资源

|  |  |  |
| --- | --- | --- |
| **学号** | **姓名** | **职责** |
| 1120161886 | 陈有贤  （组长） | 进度检控、文档总结、静态测试组 |
| 1120161927 | 彭堂智 | 文档、静态测试组 |
| 1120161888 | 范杨帆 | 文档、动态测试组 |
| 1120161935 | 王连宁 | 文档、动态测试组 |
| 1120161913 | 敬甲男 | 文档、动态测试组 |

4.4.2 测试环境

操作系统：window10

开发平台：idea

界面设计：css、html

数据库：mysql

开发语言：Java

开发环境由IDEA+PyCham+Mysql组成

4.5测试风险与评估

4.5.1 系统本身的风险

1.Bug的修复情况

2.模块功能的实现情况

3.系统整体功能的实现情况

4.代码的编写质量

6.开发人员、测试人员关于项目约定的执行情况

7.开发时间的缩短导致某些测试计划无法执行

4.5.2测试过程的风险

风险1：各个模块接口不同一

评估：不同模块的接口不同并不符合软件开发规范，统计接口不规范的模块当作BUG写入测试文档中提交。

风险2：需求变更

评估：在实际软件项目中，往往存在着在软件测试中突然改变需求的情况，但是这种情况在目前貌似不存在，故不予考虑。

风险3：人力资源不足或是发生变动

评估：保证软件测试组人员灵活稳定调动

风险4：硬件资源问题

评估：提前做好测试组硬件的统计与测试

风险5：测试时间不足

评估：组长做好进度检控，保证各项测试在规定时间点前完成。比较容易解决。

风险6:文档格式不规范

评估：文档作为这个测试工作的具现，一定要符合规范，结构清晰合理，每次文档汇总由组长进行，加强检控。

1. **测试方案**

根据组内成员的初步讨论，并结合实际情况分析，本次测试主要分为静态测试与动态测试来展开，通过两者的结合保证测试的全面性；静态测试包括文档测试、代码分析、界面分析；动态测试包括功能测试、性能测试、安全性测试、易用性测试。

文档测试：

一个软件的测试是指对该软件系统全部内容（整个软件系统以及文档）进行测试，好的文档能达到提高易用性、提高可靠性、降低技术支持费用的目的，从而提高了产品的整体质量，对软件系统的文档进行测试必不可少。

本次测试项目提供了需求文档、设计文档，在根据软件工程开发生命周期所需文档确定缺失文档的情况下，针对需求文档与设计文档展开全面测试，确保其符合国家标准或公司既定章程（在本次测试中，以软件开发课程中的文档要求为标准）。

本次文档审查规范如下：

|  |  |
| --- | --- |
| 规范编号 | 规范描述 |
| SF1001 | 内容完备 |
| SF1002 | 具有封面、目录 |
| SF1003 | 各层次段落缩进合理 |
| SF1004 | 根据内容性质具有不同且统一的字号、字体设定 |
| SF1005 | 内容逻辑清晰、不重复、不会引起歧义 |

|  |  |
| --- | --- |
| 测试目标 | 确保文档数目充足符合软件开发课程的基本文档要求 |
| 测试范围： | 需求文档、设计文档 |
| 技术： | 结合课程标准进行多层面的分析 |
| 完成标准： | 文档各方面均符合规范 |
| 测试优先级： | 高 |
| 需考虑的特殊事项： | 应对文档不全、格式非正式的情况 |

代码分析：

代码分析，又称代码走读（code review），通过词法分析、语法分析、控制流、数据流分析等技术对程序代码进行扫描，验证代码是否满足规范性、安全性、可靠性、可维护性等指标的一种代码分析技术。

本次测试通过Junit与人工复读结合的方式对代码进行分析检查，确保程序代码格式风格符合标准，由于课程官方对于代码规范无特殊规定，本测试组特形成编码规范如下：

|  |  |
| --- | --- |
| 规范编号 | 规范描述 |
| SF2001 | 变量命名不规范 |
| SF2002 | 文件命名不规范 |
| SF2003 | 函数或方法命名不规范 |
| SF2004 | 不合要求的缩进 |
| SF2005 | 无效的语句 |
| SF2006 | 缺少必要的注释 |
| SF2007 | 字符集不规范 |
| SF2008 | 文件不完整 |
| SF2009 | 引用类型错误 |
| SF2010 | 不合适的设计 |

|  |  |
| --- | --- |
| 测试目标 | 确保文档数目充足符合软件开发课程的基本文档要求 |
| 测试范围： | 被测系统的全部代码文件 |
| 技术： | 结合编程规范与测试工具进行多层面的分析 |
| 完成标准： | 文档各方面均符合规范 |
| 测试优先级： | 高 |
| 需考虑的特殊事项： | 应对文档不全、格式非正式的情况 |

界面分析：

用户界面（UI）测试用于核实用户与软件之间的交互是否恰当。UI测试的目标是确保用户有良好的交互体验。

用户界面的评测是易用性测试中很重要的一环，需要花费相对较多的经历。易用性测试用于核实软件指定条件下使用时，软件产品被理解、学习、使用和吸引用户的能力。

|  |  |
| --- | --- |
| 测试目标 | 确保用户界面简洁明了，确保用户体验良好 |
| 测试范围： | 系统中的用户界面与管理员页面 |
| 技术： | 为每个窗口创建或修改测试，以核实各个应用程序窗口和对象都可正确地进行浏览。  窗口布局符合设计规范 |
| 完成标准： | 成功地核实出各个窗口都与基准版本保持一致；在操作时，没有不便之处 |
| 测试优先级： | 中 |
| 需考虑的特殊事项： | 本项测试的重点应该放在统一性与易用性上，美观性为辅。 |

功能测试：

根据系统功能说明书，经过研讨确定最典型的用例。即从验证起草系统基本功能入手，通过多个用例的运行，来确定整个系统的功能需求是否得到了满足。此类测试基于黑盒、白盒技术，主要通过图形用户界面（GUI）与应用程序进行交互，并对交互的输出或结果进行分析，以此来核实应用程序是否恰当地完成预定的全部需求。

|  |  |
| --- | --- |
| 测试目标 | 确保测试的功能正常，其中包括登录，用户注册，查询和管理员的管理功能等。 |
| 测试范围： | 系统功能书中全部既定实现功能 |
| 技术： | 利用典型数据来执行系统，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  功能是否稳定实现 |
| 完成标准： | 系统内所有的功能均符合系统功能书中叙述的内容 |
| 测试优先级： | 高 |
| 需考虑的特殊事项： | 尤其应该关注数据库部分 |

性能测试：

性能评测是一种性能测试,对响应时间,事务处理速率和其他与时间相关的需求进行评测和评估。在本项目中主要针对数据库变更以及不同表之间的联系与调用展开测试，且确定其平均处理时间。

|  |  |
| --- | --- |
| 测试目标 | 核实所指定的事务或业务功能在下列情况下的性能行为：  预期的正常工作量下系统的反应速度与处理效率 |
| 测试范围 | 系统各个功能模块 |
| 技术 | 通过修改数据文件来增加事务数量,或通过修改脚本来增加每项事务的迭代数量 |
| 完成标准 | 单个事务或单个用户：在每个事务所预期时间范围内成功地完成测试脚本，没有发生任何故障，且处理效率处于可接受范围 |
| 测试优先级 | 中 |
| 需考虑得特殊事件 |  |

安全性测试：

安全性和访问控制测试侧重于安全性的两个关键方面：应用程序级别的安全性，即不同级别用户对数据或业务功能的访问；系统级别的安全性，即是否对sql注入等攻击手段具有一定的防御能力。

值得注意的是，应用程序级别的安全性可确保：在预期的安全性情况下，用户只能访问特定的功能或用例，或者只能访问有限的数据。例如，可能会允许所有人输入数据，创建新帐户，但只有管理员才能删除这些数据或帐户。

|  |  |
| --- | --- |
| 测试目标 | 应用程序级别的安全性：核实用户只能访问其所属用户类型已被授权访问的那些功能或数据。  系统级别的安全性：核实系统是否对攻击有一定的抵抗性 |
| 测试范围 | 主要是数据库部分 |
| 技术 | Sql注入，sql后台检控 |
| 完成标准 | 各种已知的用户类型都可访问相应的功能或数据，而且所有事务都按照预期的方式运行  对简单攻击有一定的抵御性 |
| 测试重点和优先级 | 中 |
| 需考虑得特殊事件 |  |

易用性测试：

与界面分析不同，主要从系统动态运行的情况下，开展本项测试；针对软件系统的运行易用性进行分析评测。

|  |  |
| --- | --- |
| 测试目标： | 确保产品能够易于使用，功能易于理解，操作易于控制 |
| 测试范围: | 整个系统 |
| 技术： | 黑盒组员直接使用产品或是请非测试人员（陌生人群）使用产品，对产品的各个功能进行使用，并记录感受 |
| 完成标准： | 产品易于使用；产品易于理解；产品的界面风格，操作风格比较一致；产品对用户有一定的吸引力。 |
| 测试优先级： | 高 |
| 需考虑的特殊事项： | 注意邀请多种人群参与测试，以获取全面而不同的结果 |