河北师大校园信息资讯论坛

测试计划

版本1.0

文档属性

文档属性

| 项目 | 明细 |
| --- | --- |
| 文档标题 | 河北师大校园信息资讯论坛测试计划 |
| 整理人 | 苏泽旺 |
| 创建日期 | 20200511 |
| 最后修改日期 | 20200511 |

修改记录

| 日期 | 整理人 | 版本 | 修改说明 |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

审核

| 姓名 | 认可版本 | 职位 | 日期 |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

目录

[1. 引言 4](#_Toc304268686)

[1.1 编写目的 4](#_Toc304268687)

[1.2 背景 4](#_Toc304268688)

[1.3 参考资料 4](#_Toc304268689)

[1.4 测试提交成果 4](#_Toc304268690)

[2. 测试需求 5](#_Toc304268691)

[2.1.1 测试功能模块： 5](#_Toc304268692)

[3. 测试策略 5](#_Toc304268693)

[3.1 测试类型 5](#_Toc304268694)

[3.1.1 功能测试 5](#_Toc304268695)

[3.1.2 界面/易用性测试 6](#_Toc304268696)

[3.1.3 BVT/回归测试 7](#_Toc304268697)

[3.2 版本发布策略 7](#_Toc304268698)

[3.3 阶段测试策略 7](#_Toc304268699)

[3.4 Bug管理策略 8](#_Toc304268700)

[3.4.1 缺陷管理工具 8](#_Toc304268701)

[3.4.2 错误优先级 8](#_Toc304268702)

[3.4.3 错误识别依据 8](#_Toc304268703)

[3.4.4 严重程度分类 8](#_Toc304268704)

[3.5 进度反馈策略： 9](#_Toc304268705)

[3.6 内部例会 9](#_Toc304268706)

[4. 测试环境 9](#_Toc304268707)

[5. 测试工具 9](#_Toc304268708)

[6. 通过准则 9](#_Toc304268709)

[7. 里程碑及人员分配 10](#_Toc304268710)

[细化测试任务 10](#_Toc304268711)

[8. 测试风险分析 11](#_Toc304268712)

[8.1 计划风险 11](#_Toc304268713)

# **引言**

## **编写目的**

针对河北师大校园信息资讯论坛编写本次测试计划，本文档对具体后续测试工作安排进行规划，一方面使整个项目组明确测试进度、人员分配及主要职责等；另一方面该文档定义软件测试策略、方法、范围、进度、资源等，指导测试活动的进行，使测试组成员对具体工作有更清晰的了解，按照测试计划进行后期测试工作的开展。从根本上保证系统的切实可行性。

该计划阅读对象包括：测试人员、开发人员、项目其他人员。

## **背景**

项目名称：河北师大校园信息资讯论坛

项目提出者：河北师大软件学院IT项目管理小组

开发人员：项目组

测试人员：测试组

项目目标：为河北师大学生提供服务，满足河北师大学生的需求。

## **参考资料**

《项目章程.docx》

《资源需求估计.docx》

《里程碑计划.docx》

## **测试提交成果**

《河北师大校园信息资讯论坛测试计划书.docx》

《河北师大校园信息资讯论坛测试用例.docx》

# **测试需求**

本项目计划支持Android N及iOS12以后移动设备。用户使用的功能是本需求中所定义的基础通讯功能及讨论区功能。主要面向学生，教职工，管理员三种用户。重点进行功能、界面、易用性、兼容性及性能测试。

主要测试范围如下：

移动设备客户端功能：登入登出、个人主页、消息列表、联系人列表、聊天窗口、讨论区信息流、图书馆选座系统、图书馆借阅系统、校园卡充值等功能。

功能性测试范围

### *测试功能模块：*

|  |  |  |  |
| --- | --- | --- | --- |
| 移动客户端 | 模块 | 优先级 | 说明 |
| 登入登出 | 高 | 学号、教工号、管理员账户登录 |
| 消息列表 | 高 | 个人消息，群聊信息，讨论区互动提醒，公告通知 |
| 联系人列表 | 高 | 分为群组及个人，查看名片，资料备注，发起交流 |
| 个人主页 | 高 | 包括个人信息及个人动态 |
| 聊天窗口 | 高 | 文字、图片、语音、视频、文件等多格式支援 |
| 讨论区信息流 | 高 | 关注及推荐算法 |
| 图书馆选座系统 | 中 | 接入学校图书馆选座系统 |
| 图书馆借阅系统 | 中 | 接入学校图书馆借阅系统 |
| 校园卡充值 | 中 | 接入学校校园卡充值系统 |

# **测试策略**

## **测试类型**

主要进行功能测试，功能测试之后进行界面/易用性测试，测试的过程中迭代使用BVT/回归测试。

### 功能测试

**概述：**主要验证移动设备客户端的功能是否满足《需求确认书.docx》中所规定的的功能性需求。

* **目标：**确保软件需求说明书中要求的各个功能模块全部按需求实现。

|  |  |
| --- | --- |
| **测试目标** | 各功能模块均可正常使用 |
| **测试方法和技术** | 按照测试需求、通过准则、测试用例，采用黑盒测试法，核实以下内容：  在功能正常工作时及时传达信息  在部分接口失灵时及时回报  各功能模块可以正确工作。 |
| **完成标准** | 测试计划全部执行。  无法实现的不合理需求进行需求变更。  基本功能稳定性保障 |
| **需考虑的特殊事项** | Android及iOS版本兼容性问题，由于设备类型众多，此项可能需要大量成本和时间。 |

### 界面/易用性测试

* **概述：**主要是使用户与软件之间的交互能够正常且简易地进行，用户能够满意我们开发出的项目
* **目标：**达到以下目标：
  + 各种操作的反馈都是正常的，测试完每一处功能都能正常使用；
  + 界面美观易用，经过初步调研之后，好评率较高

|  |  |
| --- | --- |
| **测试目标** | 查看界面排版、ui样式，符合师大学生这个群体的审美 支付，聊天等界面的ui符合功能，不“跑题” 点击按键之后反馈时间是否够短，响应时的ui是否会给用户带来厌烦情绪 |
| **测试方法和技术** | 根据整体架构设计,实际操作，以及界面原型检验页面 |
| **完成标准** | 测试已全部执行。  无明显瑕疵，测试团队以及调研对象的使用体验都较好 |
| **需考虑的特殊事项** | 这是一款给师大学生制作的软件，这个群体的特性是第一位的 |

### BVT/回归测试

* **概述：**当软件发生变化时，需要重新测试原来已通过测试的区域，并且验证修改的正确性。
* **目标：**验证修改后的BUG，查看是否影响了其他的功能。

|  |  |
| --- | --- |
| **测试目标** | 验证修改后的bug,如果原先的bug已经修复,就查看是否影响其他的功能 |
| **测试方法和技术** | 验证前一版本提交的bug,按照提交bug时给定的数据，在最新版本上进行验证 |
| **完成标准** | 修复bug之后，不影响之前的版本 |
| **需考虑的特殊事项** | 有些二级页面，比如点击支付之后出现的页面，也要都测试到 |

## **版本发布策略**

原则1：当首轮测试时，若系统主干功能不能通过冒烟测试，则需要开发组重新发布版本，再对新版本进行首轮测试，切莫不重新开发

原则2：遵循每日构建原则。每日构建工作由测试团队负责，每日发布新的测试版本（Build）并对其进行BVT测试，BVT测试通过后针对该Build进行细测。要求：每个成功的Build都应该通过BVT测试

## **阶段测试策略**

针对实际项目情况，测试阶段分为：

1、**单元测试阶段：**由开发人员针对自己负责的区块进行单元测试，比如支付，预约，前端页面等方面  
2、**冒烟测试阶段：**验证系统第一版本的关键功能，如果可用，则进行具体功能细测，防止出现过多bug，出现bug及时修复，再发布版本  
3、**细测阶段：**重点验证软件功能是否满足需求，由测试人员完成  
4、**BVT测试阶段：**针对新Build进行版本功能验证，通过后方可进行新功能点的细测阶段，并进行全方位测试及初步调研

5、**回归测试阶段：**重点验证bug是否解决，进行最后的市场调研，确认无误后可以上架，严格保留每个版本的信息，相关记录和代码包

## **Bug管理策略**

### *缺陷管理工具*

采用Redmine缺陷管理系统进行缺陷实时提交和跟踪。

Redmine缺陷管理系统地址：搭建未完成，需另行通知

Redmine登录名：个人的姓名全拼 如：裴军霞的用户名为 peijunxia

Redmine默认密码：个人EDU邮箱密码

### *错误优先级*

立刻级别：必须立即修改；

紧急级别：立即修改（最长时间不得超过2天）；

高级级别：在投入正式运行前修改；

普通级别：在投入正式运行前大部分需修改，未修改的缺陷需进行讨论；

低级别：可延期修改或不修改。

### *错误识别依据*

1. 需求文档
2. 技术规格说明

### *严重程度分类*

|  |  |
| --- | --- |
| **严重程度** | **现象** |
| 严重 | ①   用户需求未实现（影响到用户完成业务）；  ②   用户需求实现错误（影响到用户完成业务）；  ③   导致被测软件响应明显很慢（假死）、死机、非法退出、崩溃；  ④   导致后台数据受损或丢失 |
| 中等 | ①用户需求实现错误（不影响用户完成业务、用户使用不频繁)；  ② 用户操作过程中系统出现异常报错，但不影响系统功能的使用。  ③用户使用不频繁的功能，响应时间超出忍耐限度；  ④ UI上存在错误引导用户的信息。 |
| 轻微 | ①   UI控件不符合界面规范。  ②   影响UI友好性 |

## **进度反馈策略：**

1. 项目组内部每天以日报的形式沟通当日工作内容、进展、计划
2. 测试全部完成后，由测试管理人员向开发组反馈测试整体情况

## **内部例会**

1. 执行测试阶段开始后，测试组每天上班前进行测试的作战会议

# **测试环境**

|  |
| --- |
| **软件环境（相关软件、操作系统等）** |
| 服务器：Windows Server 2003+SQL Server 2005 +office excel+Excel Server |
| 客户端：Andriod（手机），JDK+Eclipes（pc） |
| 中间件：JDK+Eclipes |
| **硬件环境（网络、设备等）** |
| 测试服务器：尽量模拟真实运行环境 |
| 客户端：单元测试阶段使用PC的手机模拟程序，系统测试阶段使用安卓手机 |

# **测试工具**

|  |  |  |  |
| --- | --- | --- | --- |
| **用途** | **工具** | **生产厂商/自产** | **版本** |
| 缺陷跟踪 | Redmine |  |  |
| 测试用例 | Test link |  |  |

# **通过准则**

1. 所有的测试类型都执行并达到完成标准。
2. 需求覆盖率达到100%。
3. 立刻\紧急\高级别的错误修复率达到100%。
4. 普通级错误的修复率达到90%以上。
5. 严重、中等缺陷修复率100%，轻微类型缺陷个数控制在5个。

# **里程碑及人员分配**

|  |  |  |  |
| --- | --- | --- | --- |
| **阶段** | **区间测试任务** | **阶段完成标志** | **人员** |
| 测试计划  （3月20日） | 阅读需求，编写测试计划（生成V1.0），分派测试任务。 | 提交《河北师大校园信息资讯论坛测试计划》并通过评审。 | 苏泽旺 |
| 设计阶段  （3月20日—4月7日） | 熟悉系统需求并设计测试用例；测试中同步细化、更新用例。 | 提交《河北师大校园信息资讯论坛测试用例设计》。 | 测试组成员 |
| 执行功能测试  （4月15日—5月15日） | 客户端功能测试，参照开发进度开展；  同步进行客户端与中间件交互测试；  同步进行BVT及回归测试。 | 达到通过准则 | 测试组成员 |
| 测试总结报告编写  （5月15日） |  | 提交《河北师大校园信息资讯论坛总结报告》。 | 苏泽旺 |

# **细化测试任务**

|  |  |  |
| --- | --- | --- |
| **阶段** | **功能模块分配** | **人员** |
| 设计阶段  （3月20日—4月7日） | 梁栋：登录+注册  张寒崧：一卡通充值功能  孔翔宇：图书馆占座功能  苏泽旺：通讯录功能  郭佳兴：个人主页 | 测试组成员 |
| 执行功能测试  （4月15日—5月15日） | 测试组成员：客户端功能模块 | 测试组成员 |
| 测试总结报告编写  （5月15日） |  | 苏泽旺 |

# **测试风险分析**

1、需求风险

产品需求的不明确，对产品需求理解不准确，导致测试范围存在误差，遗漏部分需求或者执行了错误的测试方式；另外需求变更导致测试用例变更，测试用例维护成本增加，实时更新时存在误差。

2、测试用例风险

测试用例设计不完整，忽视了边界条件、异常输入等情况，用例覆盖率没有做到足够覆盖，测试用例没有得到全部执行，有些用例被有意或者无意的漏测，需求变更导致的测试时间被压缩等情况。

3、缺陷风险

某些缺陷偶发，难以重现，容易被遗漏；缺陷跟踪不够积极主动，没做好缺陷记录和及时更新，同样的缺陷，导致的原因可能不同，对这点没意识到导致的线上生产问题等。

4、代码质量风险

代码质量差，可读性差，重构性差，没做好注释等原因导致缺陷较多，修改难度增大；另外还有系统架构设计的不足，导致的扩展性不足，性能兼容差等问题。

5、测试环境风险

测试环境和生产环境配置不同，测试环境交叉影响较大，测试环境数据量不足导致的测试结果误差等问题。

6、测试技术风险

某些项目存在技术难度，测试能力和经验所限，技术水平相对较差导致测试进展缓慢，测试结果准确性不够，项目发布日期延期等问题。

7、回归测试风险

回归测试，一般时间相对来说较少，且大多只回归主要的功能点用例，可能造成漏测；另外还有回归验证缺陷时业务流走不通导致的打回修复再验证造成的时间延后问题。

8、沟通协调风险

项目进行过程中需要多方沟通协调，不同部门，岗位之间的沟通、协作，难免存在误解、沟通不畅的情况，比如需求变更没有及时沟通，开发代码提交没有及时告知，测试结果的反馈不及时等问题。

9、研发流程风险

其中包括从产品需求评审、研发设计、代码提交、测试发布等一些列流程，流程的不规范不协调很可能导致很多问题；比如开发在不告知其他成员的情况下提交代码，发布没有预生产环境，生产出现问题无法及时回滚等很多说烂了的情况。流程没必要一板一眼的执行，但没有流程是万万不行的。

10、其它不可预计风险

一些突发状况、不可抗力等也构成风险因素，且难以预估和避免。对于这种情况，往往一时无法解决，建议做好备份方案和容灾机制，或者采用灰度发布等措施。

PS:以上是测试过程中可能发生的风险及原因，其中有的风险是难以避免的，如缺陷风险；有的风险从理论上可以避免，如需求风险，沟通风险等；还有些风险在实际操作过程中出于时间和成本的考虑，

也难以完全回避，如回归测试风险等。

对于这些风险的存在，要尽量避免，也要做好备份方案和容灾机制，规范流程，明确职责，尽可能将风险降到可接受范围内。

## **计划风险**

1. 项目进度较紧张且项目组各成员工作安排紧密，在任务提交及时性上存在风险
2. 项目组开发进度不吻合，沟通上容易存在问题