文档编号：2-006

测试计划

项目名称：工程认证指标点计算系统

承担部门：造化钟神秀

项目负责人：鄢文哲

项目起止年限：**2018年8月27日至2018年9月15日**

文档修订

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **日期** | **更改人** | **描述（注明修改的条款或页）** |
| V1.0 | 2018.9.5 | 葛晶 | 第一版初稿，未确定内容已用批注标出。添加了更详细的测试需求分析 |

批准人签字

|  |  |  |
| --- | --- | --- |
| **职务** | **姓名** | **日期** |
| 项目经理 | 鄢文哲 | 2018.9.5 |

目录

[测试计划 I](#_Toc523949413)

[一、引言 1](#_Toc523949414)

[1. 项目背景 1](#_Toc523949415)

[2. 编写目的 1](#_Toc523949416)

[3. 环境配置 2](#_Toc523949417)

[3.1 开发环境 2](#_Toc523949418)

[3.2测试环境 2](#_Toc523949419)

[4. 参考文档 2](#_Toc523949420)

[二、测试范围 3](#_Toc523949421)

[1. 功能模块图 3](#_Toc523949422)

[2. 系统功能需求 3](#_Toc523949423)

[3. 系统性能需求 5](#_Toc523949424)

[3.1 响应时间 5](#_Toc523949425)

[3.2 事务响应时间 6](#_Toc523949426)

[3.3 适应性 6](#_Toc523949427)

[三、测试策略 7](#_Toc523949428)

[1. 功能测试 7](#_Toc523949429)

[2. 性能测试 8](#_Toc523949430)

[3. 用户界面测试 9](#_Toc523949431)

[4. 兼容性测试 10](#_Toc523949432)

[5. 安全性测试 11](#_Toc523949433)

[6. 接口测试 11](#_Toc523949434)

[六、资源 14](#_Toc523949435)

[1. 人力资源 14](#_Toc523949436)

[2. 系统资源 14](#_Toc523949437)

[七、风险评估 15](#_Toc523949438)

[1. 风险管理 15](#_Toc523949439)

[2. 变更管理 17](#_Toc523949440)

[3. 缺陷级别 17](#_Toc523949441)

[八、测试交付件 19](#_Toc523949442)

# 一、引言

## 1. 项目背景

工程认证指标点计算系统是一个用于管理和统计工程认证指标点达成度的平台，具体包括根据教师提交的文件进行原始成绩录入，然后根据一定的评价方法，得到每门课程的评价值。最后每门课程选择两年评价值的最小值，求和得到每个指标点的总评价值，判断是否达标。责任教授用户对于每年的课程系数表可以进行动态管理。本系统为教师用户和责任教授提供了更好更简洁的管理方式，提高工作效率,提供了更好更快的服务。

## 2. 编写目的

测试是软件工程中必不可少的一项，软件质量的保证必须通过测试实现，在实际工程中，测试本身就是一项耗时耗力的工程，测试本身必须要有计划才能保证再最低的成本支出下实现最优配置，保证项目的质量。制定测试计划主要出于一下几个方面的考虑：

* 为测试各项活动制定一个现实可行的、综合的计划，包括每项测试活动的对象、范围、方法、进度和预期结果。
* 为项目实施建立一个组织模型，并定义测试项目中每个角色的责任和工作内容。
* 开发有效的测试模型，尽量降低测试的难度和测试的风险。
* 确定测试所需的时间和资源，以保证其可获得性、有效性。
* 确立每个测试阶段测试完成以及测试成功的标准、要实现的目标。
* 识别处测试活动中各种风险，并消除可能存在的风险，降低由不可能消除的风险所带来的损失。

## 3. 环境配置

### 3.1 开发环境

操作系统:windows7

界面设计:css、jsp、js

数据库:MySQL

开发语言:Java

技术支持：SSM、JSP、DIV、CSS、JAVASCRIPT、JQUERY、AJAX

### 3.2测试环境

操作系统：windows10

平台：IntelliJ IEDA 2017.3.5

数据库:MySQL

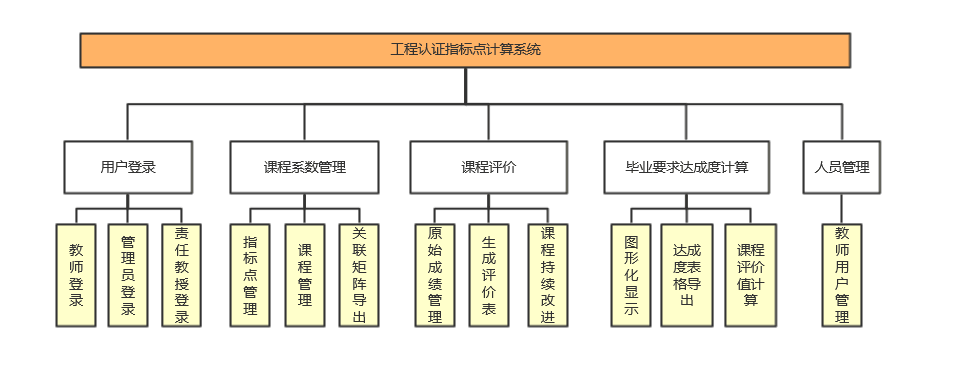
## 4. 参考文档

1. 软件测试方法和技术（第二版），朱少民主编，清华大学出版社
2. ECIPCS\_需求规格说明书\_1.1
3. 测试计划\_FBI，易泳

# 二、测试范围

工程认证指标点计算系统

## 1. 功能模块图



**图1 工程认证指标点计算系统**

## 2. 系统功能需求

|  |  |  |
| --- | --- | --- |
| 模块名称 | 功能名称 | 主要功能概述 |
| 用户登录 | 登录 | 用户角色包括：教师、管理员和责任教授。  所有用户通过用户名和密码登录系统，不同的角色有不同的权限，登录之后可以进行不同的操作。不同的用户进入系统之后的界面也会有所不同。 |
| 课程系数管理 | 指标点管理 | 责任教授可以查看当年的所有指标点，以及总的指标要求。可以根据名字和描述查询相应的指标点。还可以对课程的指标点进行修改。 |
| 课程管理 | 责任教授对每年的课程可以增删改查，在改动一个课程对支撑指标的系数值时其他的系数也会改变，要保持总和为1。 |
| 关联矩阵导出 | 责任教授可以导出指标点和课程支持系数的矩阵excel表。 |
| 课程评价 | 原始成绩管理 | 教师用户可以手动录入学生的原始成绩，也可以用excel文件自动导入。 |
|  | 评价方法管理 | 教师用户选择自己导入的原始成绩列项，再选择运算符，形成计算公式。选择已导入的成绩和计算公式来计算，生成课程评价表，主要包括该课程对相应指标点的评价值。 |
|  | 课程持续改进 | 教师用户需要对学生本学期给出改进建议。 |
| 毕业要求达成度计算 | 图形化显示 | 以柱状图的形式显示指标点的总评价值和达标基准值，显示对比，便于观看浏览； |
| 达成度表格导出 | 责任教授生成总的指标点评价值之后，可以以excel的形式导出下载 |
| 课程评价值计算 | 责任教授可以选择参与计算的教师的课程 |
| 人员管理 | 教师用户管理 | 管理员用户负责对系统人员信息的增删改查。 |

## 3. 系统性能需求

### 3.1 响应时间

响应时间＝网络响应时间+应用程序响应时间。标准可参考国外的3/5/10原则：

（1）在3秒钟之内，页面给予用户响应并有所显示，可认为是“很不错的”；

（2）在3~5秒钟内，页面给予用户响应并有所显示，可认为是“好的”；

（3）在5~10秒钟内，页面给予用户响应并有所显示，可认为是“勉强接受的”；

（4）超过10秒就让人有点不耐烦了，用户很可能不会继续等待下去。

根据项目应用场景，系统的响应时间应该在5秒种内给予用户响应。

### 3.2 事务响应时间

事务可能由一系列请求组成,事务的响应时间主要是针对用户而言,属于宏观上的概念，是为了向用户说明业务响应时间而提出的.例如:跨行取款事务的响应时间就是由一系列的请求组成的.事务响应时间是直接衡量系统性能的参数。

### 3.3 适应性

该系统采用SSM框架搭建，分成持久层、业务层和表现层，主要通过前端控制器对网页请求进行处理，它的注解十分方便，简洁化了开发和测试。需要做兼容性测试，以根据不同浏览器做一些适配。

# 三、测试策略

## 1. 功能测试

1. **链接测试**

链接是Web应用系统的一个主要特征，它是在页面之间切换和指导用户去一些不知道地址的页面的主要手段。链接测试可分为三个方面。首先，测试所有链接是否按指示的那样确实链接到了该链接的页面；其次，测试所链接的页面是否存在；最后，保证Web应用系统上没有孤立的页面，所谓孤立页面是指没有链接指向该页面，只有知道正确的URL地址才能访问。

1. **表单测试**

当用户通过表单提交信息的时候，都希望表单能正常工作。

如果使用表单来进行在线注册，要确保提交按钮能正常工作，当注册完成后应返回注册成功的消息。如果使用表单收集配送信息，应确保程序能够正确处理这些数据，最后能让顾客收到包裹。要测试这些程序，需要验证服务器能正确保存这些数据，而且后台运行的程序能正确解释和使用这些信息。

当用户使用表单进行用户注册、登陆、信息提交等操作时，我们必须测试提交操作的完整性，以校验提交给服务器的信息的正确性。例如：用户填写的出生日期与职业是否恰当，填写的所属省份与所在城市是否匹配等。如果使用了默认值，还要检验默认值的正确性。如果表单只能接受指定的某些值，则也要进行测试。例如：只能接受某些字符，测试时可以跳过这些字符，看系统是否会报错。

1. **数据校验**

如果系根据业务规则需要对用户输入进行校验，需要保证这些校验功能正常工作。例如，手机号码字段可以用一个有效字符串匹配进行校验。在这种情况下，需要有效字符串匹配完整而且程序正确运行(例如添加一个测试值，确定系统能够接受这个测试值)。

在测试表单时，该项测试和表单测试可能会有一些重复。

1. **Cookies测试**

Cookies通常用来存储用户信息和用户在某应用系统的操作，当一个用户使用Cookies访问了某一个应用系统时，Web服务器将发送关于用户的信息，把该信息以Cookies的形式存储在客户端计算机上，这可用来创建动态和自定义页面或者存储登陆等信息。

1. **数据库测试**

数据库为Web应用系统的管理、运行、查询和实现用户对数据存储的请求等提供空间。在Web应用中，最常用的数据库类型是关系型数据库，可以使用SQL对信息进行处理。

在使用了数据库的Web应用系统中，一般情况下，可能发生两种错误，分别是数据一致性错误和输出错误。数据一致性错误主要是由于用户提交的表单信息不正确而造成的，而输出错误主要是由于网络速度或程序设计问题等引起的，针对这两种情况，可分别进行测试。

## 2. 性能测试

1. **连接速度测试**

用户连接到Web应用系统的速度根据上网方式的变化而变化，他们或许是电话拨号，或是宽带上网。当下载一个程序时，用户可以等较长的时间，但如果仅仅访问一个页面就不会这样。如果Web系统响应时间太长（例如超过5秒钟），用户就会因没有耐心等待而离开。

## 3. 用户界面测试

界面是软件与用户交互的最直接的层，界面的好坏决定用户对软件的第一印象。而且设计良好的界面能够引导用户自己完成相应的操作，起到向导的作用。同时界面如同人的面孔，具有吸引用户的直接优势。设计合理的界面能给用户带来轻松愉悦的感受和成功的感觉，相反由于界面设计的失败，让用户有挫败感，再实用强大的功能都可能在用户的畏惧与放弃中付诸东流。目前界面的设计引起软件设计人员的重视的程度还远远不够，直到最近网页制作的兴起，才受到专家的青睐。而且设计良好的界面由于需要具有艺术美的天赋而遭拒绝。

主要分以下几个方面进行：

* 用户界面的友好性：用户界面做到布局合理、简单易用，让用户能够轻松理解每个功能的使用，操作方便，与需求规格说明书规定的功能需求一致。
* 易操作性：该软件针对广大的消费群体，用户面广，用户对电脑的了解及实用程度有强有弱，软件要针对这一消费特征设计易学习的操作流程，让消费者轻松上手。
* 界面风格一致性：用户界面的风格和操作是否一致
* 错误信息提示：用户做了一些不合理的操作，例如输入字符串与规定字符不符合时，界面出现友好的、正确的提示信息。

## 4. 兼容性测试

1. **平台测试**

市场上有很多不同的操作系统类型，最常见的有Windows、Unix、Macintosh、Linux等。Web应用系统的最终用户究竟使用哪一种操作系统，取决于用户系统的配置。这样，就可能会发生兼容性问题，同一个应用可能在某些操作系统下能正常运行，但在另外的操作系统下可能会运行失败。

因此，在Web系统发布之前，需要在各种操作系统下对Web系统进行兼容性测试。

1. **浏览器测试**

浏览器是Web客户端最核心的构件，来自不同厂商的浏览器对Java，、JavaScript、 ActiveX、 plug-ins或不同的HTML规格有不同的支持。例如，ActiveX是Microsoft的产品，是为Internet Explorer而设计的，JavaScript是Netscape的产品，Java是Sun的产品等等。另外，框架和层次结构风格在不同的浏览器中也有不同的显示，甚至根本不显示。不同的浏览器对安全性和Java的设置也不一样。

测试浏览器兼容性的一个方法是创建一个兼容性矩阵。在这个矩阵中，测试不同厂商、不同版本的浏览器对某些构件和设置的适应性。

1. **分辨率测试**

页面版式在 640x400、600x800 或 1024x768 的分辨率模式下是否显示正常? 字体是否太小以至于无法浏览? 或者是太大? 文本和图片是否对齐?

## 5. 安全性测试

主要是测试系统在没有授权的情况下，内部或者外部用户对系统进行攻击或者恶意破坏时如何进行处理，是否仍能保证数据的安全。

项目“基于SSM的某酒店客房管理系统”作为网上酒店预订系统，需要用户进行登录，以验证他们的身份。这样对用户是方便的，他们不需要每次都输入个人资料。需要验证系统阻止非法的用户名/口令登录，而能够通过有效登录。用户登录是否有次数限制?是否限制从某些IP地址登录?如果允许登录失败的次数为3，你在第三次登录的时候输入正确的用户名和口令，能通过验证吗?口令选择有规则限制吗?

是否可以不登陆而直接浏览某个页面？

Web应用系统是否有超时的限制，也就是说，用户登陆后在一定时间内（例如15分钟）没有点击任何页面，是否需要重新登陆才能正常使用。

## 6. 接口测试

1. **服务器测试**

本次项目为了减少环境配置的麻烦，测试之前已经将一名组员的电脑当作服务器运行着该项目，其他人只需要通过ip+端口就能访问。

需要测试的接口是浏览器与服务器的接口。测试人员提交事务，然后查看服务器记录，并验证在浏览器上看到的正好是服务器上发生的。测试人员还可以查询数据库，确认事务数据已正确保存。

1. **错误处理**

通常我们试图确认系统能够处理所有错误，但却无法预期系统所有可能的错误。尝试在处理过程中中断事务，看看会发生什么情况？订单是否完成？尝试中断用户到服务器的网络连接。在这些情况下，系统能否正确处理这些错误？如果用户自己中断事务处理，在订单已保存而用户没有返回网站确认的时候，需要由客户代表致电用户进行订单确认。

# 六、资源

## 1. 人力资源

本次测试组课堂分组情况如下：

测试项目组长：葛晶

测试人员：吴桐、王林楠

## 2. 系统资源

**软件资源**

操作系统：windows10 教育版

数据库：Mysql5.5

应用软件：Navicat for MySQL、IntelliJ IDEA

**硬件资源**

电脑配置：CPU intel i5 内存 8G 显卡 GTX850M 硬盘 1T

配置管理工具：git

# 七、风险评估

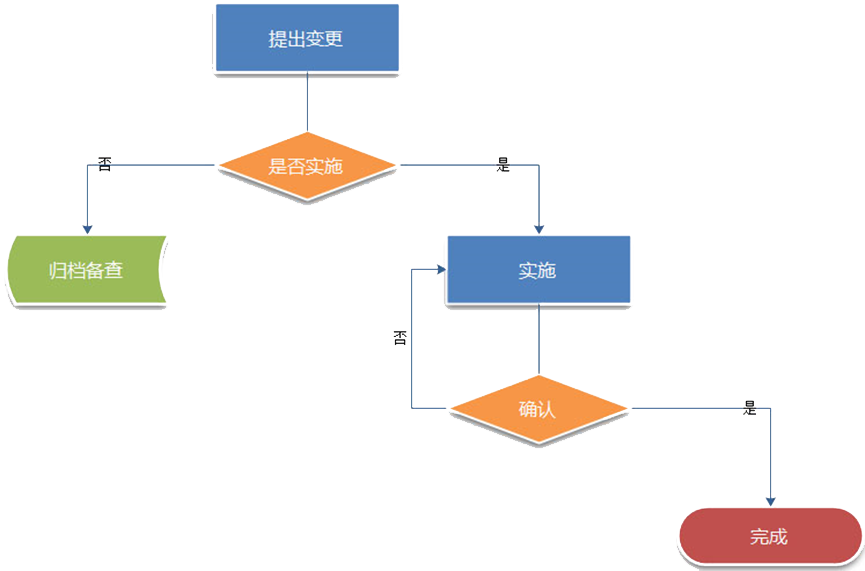
## 1. 风险管理

|  |  |  |  |
| --- | --- | --- | --- |
| 风险  来源 | 风险描述 | 风险影响 | 应对策略 |
| 范围  风险 | 与范围变更有关的风险，例如用户的需求变化，测试工作往往要重新开始 | 对产品需求缺少认同；在需求分析过程中客户参与不够；由于不确定的需求导致新的市场需求；缺少有效的需求变化管理过程；对需求变化缺少相关分析 | 计划响应需求变更；接受需求变更，并且做出相应的计划工作量等调整 |
| 进度  风险 | 由于多种因素导致测试进度无法对应测试计划上安排的时间节点，导致测试进度延期 | 可能导致整个项目的管理成本和人力成本剧增，并且存在降低测试质量的潜在危险 | 测试进度的安排要尽量详细，具体到负责人；增加计划评审环节；加强团队测试工作效率；加强风险监管措施，及时进行风险管控；与项目开发方进行协商合理安排项目测试进度 |
| 质量  风险 | 用户要求系统保证高质量，系统测试的强度和测试方案要相应的调整和完善 | 由于业务需求，用户要求提升测试的质量 | 接受  制定应急计划并在风险发生时执行，应急计划可以大大减少处理问题的费用。 |
| 技术  风险 | 是指由于与项目研制相关的技术因素的变化而给项目建设带来的风险，包括潜在的设计、实现、接口、验证和维护、技术的不确定性、“老”技术与“新”技术等方面的问题 | 更改测试计划，不确定的部分让测试人员做好充分的准备 | 需求再确认，获取更详细的信息，增派专家不确定的技术避免使用 |

## 2. 变更管理

变更管理包括需求变更、测试案例、测试报告、测试日志等变更。

要求规范各类文档的样式和变更说明规范，所有正式的关键性的变更都需要进行变更申请，变更流程如下：



**图2 变更流程**

## 3. 缺陷级别

|  |  |  |
| --- | --- | --- |
| 缺陷级别 | 缺陷描述 | 处理方案 |
| 提示 | 一些GUI问题或者友好性问题，如错误提示信息不合逻辑但不影响用户操作 | 及时记录在测试报告中，并且表明问题的描述和截图等信息，与其他错误情况一起提交 |
| 一般 | 某个bug发生了，只影响一个功能，而其他功能可以正常运行 | 综合记录之后立马上报给开发人员进行修改 |
| 严重 | 产品中使功能无法实现的BUG，比如某个功能无法运行，GUI长时间僵死没有响应 | 一旦发生这样的缺陷，必须进行详细的记录，并且立刻上报给项目组长和项目经理，项目经理立马联系系统架构师及相关人员做修复 |
| 致命 | 致命是严重影响产品的BUG，比如操作手册的错误，需求的错误等。 | 一旦发生这样的缺陷，必须进行详细的记录，并且立刻上报给项目经理，项目经理立马联系系统架构师及相关人员做修复，必要时可能更改整个项目的计划和实施过程 |

# 八、测试交付件

ECIPCS\_测试计划\_1.0.docx

ECIPCS\_测试报告\_1.0.docx