|  |
| --- |
| 学生信息管理系统  **项目测试计划** |

|  |  |  |
| --- | --- | --- |
| 文件状态：[] 草稿 [√]正式发布 []正在修改 | 文件标识 | 学生信息管理系统-项目测试计划-2014-7-6 |
| 当前版本 | v1.0.1 |
| 作者 | 陈龙龙 |
| 完成日期 | 2014年7月6日 |

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 内容/备注 | 作者 |
| 2014-07-05 | v1.0.0 | 完成初稿 | 陈龙龙 |
| 2014-07-06 | V1.0.1 | 修改定稿 | 陈龙龙 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1引言 3](#_Toc392545048)

[1.1 编写目的 3](#_Toc392545049)

[1.2 背景 3](#_Toc392545050)

[1.2.1项目的提出者 3](#_Toc392545051)

[1.2.2 项目的开发者 3](#_Toc392545052)

[1.3定义 3](#_Toc392545053)

[1.4参考资料 4](#_Toc392545054)

[2任务概述 4](#_Toc392545055)

[2.1 目标 4](#_Toc392545056)

[2.2 运行环境 5](#_Toc392545057)

[2.3 需求概述 5](#_Toc392545058)

[2.4 条件与限制 5](#_Toc392545059)

[3计划 5](#_Toc392545060)

[3.1 测试方案 5](#_Toc392545061)

[3.2 测试项目 5](#_Toc392545062)

[3.3 测试准备 6](#_Toc392545063)

[3.4 测试机构及人员 6](#_Toc392545064)

[4测试项目说明 6](#_Toc392545065)

[4.1 测试项目名称及测试内容 6](#_Toc392545066)

[4.2 测试用例 7](#_Toc392545067)

[5评价 8](#_Toc392545068)

[5.1 范围 8](#_Toc392545069)

[3.2 准则 8](#_Toc392545070)

# 1引言

## 1.1 编写目的

在需求分析阶段已大致确定本学生信息管理系统的模块，同时每个模块都有其特有的用例，而测试计划中最主要的部分即为测试用例，测试用例依靠用例而建，在此基础上，测试人员可根据测试用例对系统模块进行测试，并记录缺陷日志，使运维人员能够根据缺陷记录进行系统的改进，使系统更加完善。在软件需求说明书中第二和第三节已给出用户信息维护模块的用例详情，下面将以该用例为例进行描述。

## 1.2 背景

## 1.2.1项目的提出者

上海电力学院计算机科学与技术学院

## 1.2.2 项目的开发者

凤翔、陈龙龙

## 1.3定义

学生信息管理系统（Student Information Management System）：为用户提供充足的信息和快捷的查询手段。能有效的帮助学校和老师掌握学生的情况，为学生提供成绩跟课程方面的查询。

Node.js： 是一个基于Chrome JavaScript 运行时建立的一个平台， 用来方便地搭建快速的， 易于扩展的网络应用· Node.js 借助[事件驱动](http://baike.baidu.com/view/536048.htm" \t "_blank)， 非阻塞[I/O](http://baike.baidu.com/subview/300881/11169495.htm" \t "_blank) 模型变得轻量和高效， 非常适合 运行在分布式设备 的 数据密集型 的实时应用

MongoDB：一个基于分布式文件存储的数据库。由C++语言编写。旨在为WEB应用提供可扩展的高性能数据存储解决方案。

CoffeeScript：CoffeeScript是一套JavaScript的转译语言。受到Ruby、Python与Haskell等语言的启发，CoffeeScript增强了JavaScript的简洁性与可读性。此外，CoffeeScript也新增了更复杂的功能，例如列表内涵（List comprehension）、模式匹配（Pattern matching）等。

## 1.4参考资料

[Node.js v0.10.29 Manual & Documentation](http://nodejs.org/api/)

[The MongoDB 2.6 Manual](http://docs.mongodb.org/manual/)

[Mongoose Schemas v3.8.12](http://mongoosejs.com/docs/guide.html)

[NODE.JS UNIT TESTING BY @朴灵](http://html5ify.com/unittesting/slides/index.html#/)

[Express 4.x API Reference](http://expressjs.com/4x/api.html)

[How to use templates with EJS]( https://code.google.com/p/embeddedjavascript/wiki/Templates)

[Bootstrap 中文文档](http://v3.bootcss.com/)

[LESS 语法](http://www.bootcss.com/p/lesscss/)

[Pro Git](http://git-scm.com/book/zh/)

[初识 mocha in NodeJS](http://cnodejs.org/topic/516526766d38277306c7d277)

[Mocha - the fun, simple, flexible JavaScript test framework](http://visionmedia.github.io/mocha/)

[Wikipedia](http://zh.wikipedia.org/)

[Stackoverflow](http://stackoverflow.com/)

本项目可行性研究报告

本项目开发计划

本项目需求规格分析说明书

本项目概要设计规格说明书

本项目详细设计规格说明书

# 2任务概述

## 2.1 目标

对于开发的学生信息管理系统，主要实现以下功能：对学生的基本信息、课程安排、成绩、选课、密码，进行基本的管理。包括：学生的基本信息和学科成绩的存储、输入、删除、修改、查询。老师可以录入成绩、查询学生课程信息及修改密码、编辑学生的信息。管理员：课程管理、学生管理、修改密码及查询。

## 2.2 运行环境

本系统全部使用开源技术，可以在Windows、Mac OS X、Linux全平台通用。

## 2.3 需求概述

详见第三章项目需求规格分析说明书中第二节。

## 2.4 条件与限制

本项目受时间所局限，因为从开始至结束仅有九天时间，九天时间内单单依靠小组2人就要做完可行性研究分析报告、需求分析报告书、系统计划开发任务书、概要设计说明书、详细设计说明书、软件测试分析报告、编码工作、实习报告压力非常大，所以只能将系统大致功能完成，可能无法达到用户更高一层次的需求。另外，我们将采用很多新一点的技术，所以也有很多的技术局限性，可能并不能将本系统做的很完整。

# 3计划

## 3.1 测试方案

## 3.2 测试项目

由软件需求说明书中的3.1节我们可知，就用户信息维护这个简单的模块而言，已达5个用例描述，其他更复杂的模块的用例肯定超过五个，甚至多于十个。下面我将以登录模块和用户信息维护模块为例，列出所要测试的用例，如下表所示。

|  |  |  |
| --- | --- | --- |
| 编号 | 模块名称 | 测试用例名称 |
| 1 | 学生基本信息管理模块 | 1. 添加学生信息测试用例 2. 修改学生信息测试用例 3. 删除学生测试用例 |
| 2 | 课程信息管理模块 | 1. 添加课程信息测试用例 2. 修改课程信息测试用例 3. 删除课程测试用例 |
| 3 | 查询模块 | 1. 查看学生信息测试用例 2. 查看课程信息测试用例 3. 查看成绩信息测试用例 |
| 4 | 成绩管理模块 | 1. 添加成绩信息测试用例 2. 修改成绩信息测试用例 3. 删除成绩测试用例 |
| 5 | 统计模块 | 1. 计算平均分测试用例 2. 计算总分测试用例 3. 按区间统计测试用例 |

具体每个测试用例的内容，例如被测用例的功能、输入和输出等质量指标，详见如下第四节内容。

## 3.3 测试准备

做好人员分工、硬件设施准备齐全

## 3.4 测试机构及人员

上海电力学院计算机科学与技术学院学生信息管理系统开发小组

凤翔、陈龙龙

# 4测试项目说明

## 4.1 测试项目名称及测试内容

因时间有限，下面我将以学生基本信息管理模块中的用例1（添加学生信息）为例，建立图标，指出被测用例的功能、输入和输出等质量指标，如下表所示。

|  |  |
| --- | --- |
| 测试编号 | 001 |
| 测试目标 | 验证系统是否对点击【添加学生】按钮时做出正确的响应 |
| 测试环境 | windows 7操作系统和浏览器google chrome |
| 测试步骤 | 1. 打开浏览器，管理员登录成功 2. 跳转至学生基本信息管理模块，显示学生列表界面 3. 点击【添加学生】按钮，检查是否显示添加界面 |
| 期望结果 | 显示用户信息添加界面 |

|  |  |
| --- | --- |
| 测试编号 | 002 |
| 测试目标 | 验证学生信息未填写完整时，系统是否对【确定】按钮做出正确的响应 |
| 测试环境 | windows 7操作系统和浏览器google chrome |
| 测试步骤 | 1. 打开浏览器，管理员登录成功 2. 跳转至学生基本信息管理模块，显示学生列表界面 3. 点击【添加学生】按钮，检查是否显示添加界面 4. 输入不完整的用户信息，单击【确定】按钮 |
| 期望结果 | 系统提示“XXX信息未输入，请返回重新输入” |

|  |  |
| --- | --- |
| 测试编号 | 003 |
| 测试目标 | 验证学生信息填写错误时，系统是否对【确定】按钮做出正确的响应 |
| 测试环境 | windows 7操作系统和浏览器google chrome |
| 测试步骤 | 1. 打开浏览器，管理员登录成功 2. 跳转至学生基本信息管理模块，显示学生列表界面 3. 点击【添加学生】按钮，检查是否显示添加界面 4. 输入错误的学生信息，单击【确定】按钮 |
| 期望结果 | 系统提示“XXX信息输入有误，请重新输入” |

|  |  |
| --- | --- |
| 测试编号 | 004 |
| 测试目标 | 验证用户信息填写完全正确时，系统是否对【确定】按钮做出正确的响应 |
| 测试环境 | windows 7操作系统和浏览器google chrome |
| 测试步骤 | 1. 打开浏览器，管理员登录成功 2. 跳转至学生基本信息管理模块，显示学生列表界面 3. 点击【添加学生】按钮，检查是否显示添加界面 4. 输入正确的用户信息，单击【确定】按钮 |
| 期望结果 | 用户添加成功，并返回用户列表界面 |

## 4.2 测试用例

由4.1中的测试用例可知，该测试用例共有4个测试内容，故根据四个测试内容的输入和输出动作给出如下测试详情。

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 1 | | |
| 模块名称 | 学生基本信息管理模块 | | |
| 用例名称 | 添加学生信息用例 | | |
| 测试用例名称 | 添加学生信息测试用例 | | |
| 功能描述 | 为系统添加学生 | | |
| 编号 | 输入/动作 | 期望的输出/相应 | 实际情况 |
| 001 | 管理员点击用户【添加学生】按钮 | 系统显示学生信息添加界面 |  |
| 002 | 未将学生信息填写完整时，单击【确定】按钮 | 系统提示“XXX信息未输入，请返回重新输入” |  |
| 003 | 学生信息输入有误时，单击【确定】按钮 | 系统提示“XXX信息输入有误，请重新输入” |  |
| 004 | 所有信息填写完整且正确时，单击【确定】按钮 | 学生添加成功，并返回用户列表界面 |  |

# 5评价

## 5.1 范围

由上述测试计划我们可知，本系统的测试用例范围涵盖模块的各基本功能，但因时间限制，故给出的测试用例并不十分全面。但在此基础上，我们可根据已有的模块编写用例，再根据已有的用例编写测试用例，最后能涵盖系统的各个基本功能。

## 3.2 准则

由4.1、4.2节可知，将测试用例数据放置在表格中，不仅可以使读者能简单明了的了解该测试用例，而且还能是测试员快速的根据测试用例进行用例测试，这大大加快了系统的开发素的，是一个较好的开发流程模块之一。