|  |
| --- |
| 学生信息管理系统  **项目详细设计规格说明书** |

|  |  |  |
| --- | --- | --- |
| 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-项目详细设计规格说明书-2014-7-4 |
| 当前版本 | v1.0.1 |
| 作者 | 陈龙龙、凤翔 |
| 完成日期 | 2014年7月4日 |

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 内容/备注 | 作者 |
| 2014-07-03 | v1.0.0 | 完成初稿 | 陈龙龙 |
| 2014-07-04 | V1.0.1 | 修改定稿 | 陈龙龙 凤翔 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1引言 3](#_Toc392244190)

[1.1 编写目的 3](#_Toc392244191)

[1.2 背景 3](#_Toc392244192)

[1.2.1项目的提出者 3](#_Toc392244193)

[1.2.2 项目的开发者 3](#_Toc392244194)

[1.3定义 3](#_Toc392244195)

[1.4参考资料 4](#_Toc392244196)

[2总体设计 4](#_Toc392244197)

[2.1 需求概述 4](#_Toc392244198)

[2.2 软件结构 4](#_Toc392244199)

[3程序描述 4](#_Toc392244200)

[4数据库设计 9](#_Toc392244201)

[4.1 概念结构设计 9](#_Toc392244202)

[4.2 物理结构设计 10](#_Toc392244203)

[4.3 数据模型图设计 11](#_Toc392244204)

# 1引言

## 1.1 编写目的

本学生信息管理系统详细设计说明书使用户通过详细设计说明了解系统代码情况和系统运行流程情况。

## 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

# 2总体设计

## 2.1 需求概述

详见第三章项目需求规格分析说明书中第二节。

## 2.2 软件结构

# 3程序描述

学生基本信息管理模块详情请见第三章需求规格说明书第三节

* 课程信息管理模块

|  |  |
| --- | --- |
| 用例名称 | 添加课程信息 |
| 用例编号 | 1 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何为系统添加课程的过程 |
| 前置条件 | 1. 管理员登录系统成功 2. 输入正确的课程信息 |
| 后置条件 | 学生添加成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【课程信息维护】按钮 4. 系统显示课程列表页面 5. 单击【添加】按钮，系统显示课程添加界面 6. 输入正确的课程信息后，点击【确认】 7. 系统提示课程添加成功 8. 系统返回课程列表界面 |
| 替代流程 | 任何时刻发生以下状况，添加用户信息用例将失败：   1. 管理员未输入其中任一一项课程信息 2. 系统提示“XXX信息未输入，请返回重新输入” 3. 管理员输入错误的用户信息 4. 系统提示“XXX信息输入有误，请重新输入” |

|  |  |
| --- | --- |
| 用例名称 | 修改课程信息 |
| 用例编号 | 2 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何为修改课程信息的过程 |
| 前置条件 | 1. 管理员登录系统成功 2. 输入正确的课程信息 |
| 后置条件 | 课程信息修改成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【课程信息维护】按钮 4. 系统显示课程列表页面 5. 单击任意一名课程，系统显示该课程信息界面 6. 单击【编辑】按钮，系统进入编辑界面 7. 输入正确的课程信息后，点击【确认】 8. 系统提示课程信息修改成功 9. 系统返回课程列表界面 |
| 替代流程 | 任何时刻发生以下状况，修改课程信息用例将失败：   1. 管理员输入错误的课程信息 2. 系统提示“XXX信息输入有误，请重新输入” |

|  |  |
| --- | --- |
| 用例名称 | 删除课程 |
| 用例编号 | 3 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何删除课程的过程 |
| 前置条件 | 1. 管理员登录系统成功 |
| 后置条件 | 课程删除成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【课程信息维护】按钮 4. 系统显示课程列表页面 5. 单击任意一名课程后的【删除】按钮 6. 系统提示课程删除成功 7. 系统返回课程列表界面 |
| 替代流程 | 无 |

* 成绩管理模块

|  |  |
| --- | --- |
| 用例名称 | 录入成绩信息 |
| 用例编号 | 1 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何为学生录入成绩的过程 |
| 前置条件 | 1.管理员登录系统成功 |
| 后置条件 | 成绩录入成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生成绩维护】按钮 4. 系统显示学生成绩列表页面 5. 单击【录入成绩】按钮，系统显示录入成绩界面 6. 录入成绩后，点击【确认】 7. 系统提示成绩录入成功 8. 系统返回学生列表界面 |
| 替代流程 | 任何时刻发生以下状况，录入成绩将失败：  管理员未录入成绩信息  1）系统提示“XXX成绩未输入，请返回重新输入”   1. 系统提示“XXX成绩输入有误，请重新输入” |

|  |  |
| --- | --- |
| 用例名称 | 修改成绩信息 |
| 用例编号 | 2 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何修改学生成绩信息的过程 |
| 前置条件 | 1.管理员登录系统成功 |
| 后置条件 | 课程信息修改成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生成绩维护】按钮 4. 系统显示学生成绩列表页面 5. 单击任意一名学生的【修改成绩】按钮，系统显示该修改成绩界面 6. 输入正确的成绩信息后，点击【确认】 7. 系统提示成绩修改成功 8. 系统返回学生成绩列表界面 |
| 替代流程 | 任何时刻发生以下状况，修改课程信息用例将失败：   1. 管理员输入错误的成绩信息 2. 系统提示“XXX成绩输入有误，请重新输入” |

|  |  |
| --- | --- |
| 用例名称 | 删除成绩 |
| 用例编号 | 3 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何删除成绩信息的过程 |
| 前置条件 | 管理员登录系统成功 |
| 后置条件 | 成绩删除成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生成绩维护】按钮 4. 系统显示学生成绩列表页面 5. 单击任意一项成绩后的【删除】按钮 6. 系统提示成绩删除成功 7. 系统返回学生成绩列表界面 |
| 替代流程 | 无 |

* 统计模块

|  |  |
| --- | --- |
| 用例名称 | 按成绩区间统计 |
| 用例编号 | 1 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何按成绩区间统计学生成绩的过程 |
| 前置条件 | 1.管理员登录系统成功 |
| 后置条件 | 统计成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生成绩维护】按钮 4. 系统显示学生成绩列表页面 5. 单击【按成绩区间统计】按钮，系统显示统计界面 6. 输入正确的成绩区间后，点击【确认】 7. 系统提示成绩统计完成 8. 系统返回统计结果页面 |
| 替代流程 | 任何时刻发生以下状况，修改课程信息用例将失败：   1. 管理员输入错误的成绩区间 2. 系统提示“输入有误，请重新输入” |

* 查询模块

|  |  |
| --- | --- |
| 用例名称 | 查看学生信息 |
| 用例编号 | 1 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何查看学生信息的过程 |
| 前置条件 | 管理员登录系统成功 |
| 后置条件 | 学生信息查看成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生信息维护】按钮 4. 系统显示学生列表页面 5. 单击任意一名学生，系统显示学生信息界面 |
| 替代流程 | 任何时刻发生以下状况，查看学生信息用例将失败：   1. 管理员点击错误的按钮 2. 系统停留在原界面不跳转 3. 系统跳转至其它页面 |

|  |  |
| --- | --- |
| 用例名称 | 查看课程信息 |
| 用例编号 | 2 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何查看课程信息的过程 |
| 前置条件 | 管理员登录系统成功 |
| 后置条件 | 课程信息查看成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【课程信息维护】按钮 4. 系统显示课程列表页面 5. 单击任意一名课程，系统显示课程信息界面 |
| 替代流程 | 任何时刻发生以下状况，查看课程信息用例将失败：  管理员点击错误的按钮   1. 系统停留在原界面不跳转 2. 系统跳转至其它页面 |

|  |  |
| --- | --- |
| 用例名称 | 查看学生选课信息 |
| 用例编号 | 3 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员如何查看学生选课信息的过程 |
| 前置条件 | 1.管理员登录系统成功  2.管理员输入正确学生信息 |
| 后置条件 | 学生选课信息查看成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生信息维护】按钮 4. 系统显示学生列表页面 5. 管理员点击任意一名学生后的【选课情况】按钮 6. 系统显示该学生选课信息界面 |
| 替代流程 | 任何时刻发生以下状况，查看课程信息用例将失败：  管理员点击错误的按钮   1. 系统停留在原界面不跳转 2. 系统跳转至其它页面 |

|  |  |
| --- | --- |
| 用例名称 | 模糊查询 |
| 用例编号 | 4 |
| 执行者 | 管理员 |
| 用例简述 | 该用例描述了管理员根据学生的已知条件来查询学生的详细信息 |
| 前置条件 | 管理员登录系统成功 |
| 后置条件 | 学生信息查看成功 |
| 基本流程 | 1. 管理员登录 2. 系统显示登录后首页 3. 管理员点击【学生信息维护】按钮 4. 管理员点击【模糊查询】按钮 5. 管理员输入姓名、学号、班级、系名等已知信息，点击【确定】按钮进行模糊查询 6. 系统显示查询到的学生信息界面 |
| 替代流程 | 任何时刻发生以下状况，查看学生信息用例将失败：  管理员点击错误的按钮   1. 系统停留在原界面不跳转 2. 系统跳转至其它页面 |

# 4数据库设计

## 4.1 概念结构设计

本系统共分为如下五个模块，其分布如下所示。



## 4.2 物理结构设计

学生信息管理系统部分E-R图如下图所示。



## 4.3 数据模型图设计

