|  |
| --- |
| 学生信息管理系统  **项目概要设计规格说明书** |

|  |  |  |
| --- | --- | --- |
| 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-项目概要设计规格说明书-2014-7-3 |
| 当前版本 | v1.0.1 |
| 作者 | 凤翔、陈龙龙 |
| 完成日期 | 2014年7月3日 |

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 内容/备注 | 作者 |
| 2014-07-02 | v1.0.0 | 完成初稿 | 凤翔 |
| 2014-07-03 | V1.0.1 | 修改定稿 | 陈龙龙 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1引言 3](#_Toc392244153)

[1.1 编写目的 3](#_Toc392244154)

[1.2 背景 3](#_Toc392244155)

[1.2.1项目的提出者 3](#_Toc392244156)

[1.2.2与其他软件、系统的关系： 3](#_Toc392244157)

[1.3定义 3](#_Toc392244158)

[1.4参考资料 4](#_Toc392244159)

[2任务概述 4](#_Toc392244160)

[2.1 目标 4](#_Toc392244161)

[2.2 运行环境 5](#_Toc392244162)

[2.3 需求概述 5](#_Toc392244163)

[2.4 条件与限制 5](#_Toc392244164)

[3总体设计 5](#_Toc392244165)

[3.1 处理流程 5](#_Toc392244166)

[3.2 总体结构和模块外部设计 6](#_Toc392244167)

[3.3 功能分配 6](#_Toc392244168)

[4接口设计 7](#_Toc392244169)

[4.1 外部接口 7](#_Toc392244170)

[4.2 内部接口 7](#_Toc392244171)

[5数据结构设计 7](#_Toc392244172)

[5.1 逻辑结构设计 7](#_Toc392244173)

[5.2 物理结构设计 7](#_Toc392244174)

[5.3 数据结构与程序的关系 8](#_Toc392244175)

# 1引言

## 1.1 编写目的

本学生信息管理系统概要设计说明书可以为测试、开发以及接下来的文档做指导作用。

## 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与其他软件、系统的关系：

系统：Microsoft Windows, GNU/Linux, Mac OS X

服务器环境：Node.js

后台语言：CoffeeScript

数据库：MongoDB

IDE：JetBrains Webstorm 8

版本控制软件：Git  
编辑器：Sublime Text 3

## 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

# 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

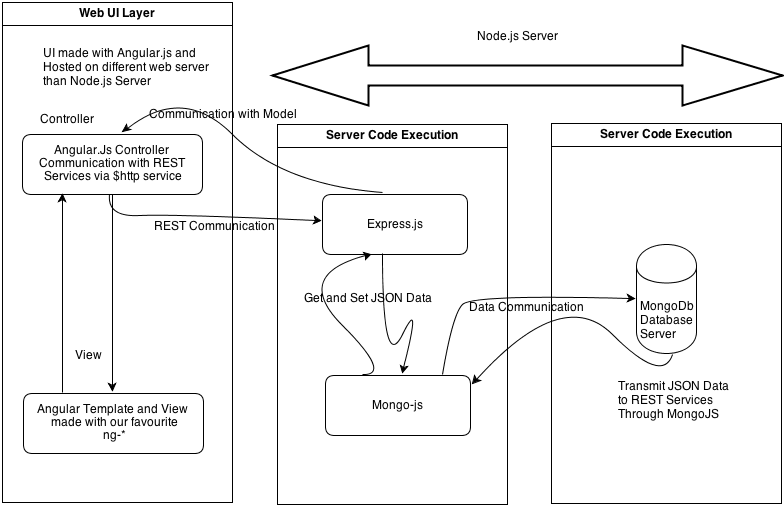
详见第三章项目需求规格分析说明书中第二节。

## 2.4 条件与限制

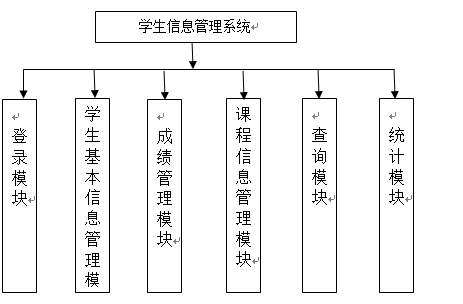
本项目受时间所局限，因为从开始至结束仅有九天时间，九天时间内单单依靠小组2人就要做完可行性研究分析报告、需求分析报告书、系统计划开发任务书、概要设计说明书、详细设计说明书、软件测试分析报告、编码工作、实习报告压力非常大，所以只能将系统大致功能完成，可能无法达到用户更高一层次的需求。另外，我们将采用很多新一点的技术，所以也有很多的技术局限性，可能并不能将本系统做的很完整。

# 3总体设计

## 3.1 处理流程



## 3.2 总体结构和模块外部设计



## 3.3 功能分配

（1）学生基本信息管理模块：对学生的基本信息进行综合管理，可以添加、修改及删除学生的基本信息。

（2）成绩管理模块：对学生所选课程的成绩信息进行综合管理，可以添加、修改及删除基本信息。

（3）课程信息管理模块：对课程信息进行综合管理，可以添加、修改及删除课程的基本信息。

（4）查询模块：

1）学生基本信息的查询：根据学生的已知条件来查询学生的详细信息，对姓名、学号、班级、系名等支持模糊查询。

2）课程基本信息的查询：根据课程的信息来查询课程的详细信息。

3）查询学生的选课情况、查询学生所选课程的成绩。

（5）统计模块：根据不同课程对学生成绩进行统计，求平均分、总分等；根据不同的分数区间进行人数统计等。

# 4接口设计

## 4.1 外部接口

本项目是B/S架构，浏览器通过80端口访问，不提供外部直接API。

## 4.2 内部接口

内部接口采用RESTful Web API，传输资源为JSON。

# 5数据结构设计

## 5.1 逻辑结构设计

全部采用JSON格式。

## 5.2 物理结构设计

Javascript内部采用JSON作为存储格式，数据库内采用BSON作为内部存储格式

## 5.3 数据结构与程序的关系

服务器通过ORM框架映射将内部的JSON存取为数据库中的BSON。