# **学生选课系统 V0.3**

目录

**[第一章 绪论](#_Toc1826654317_WPSOffice_Level1)** **[3](#_Toc1826654317_WPSOffice_Level1)**

[1.1. 目的及意义](#_Toc1629406102_WPSOffice_Level2) [3](#_Toc1629406102_WPSOffice_Level2)

**[第二章 开发工具及关键技术介绍](#_Toc1629406102_WPSOffice_Level1)** **[4](#_Toc1629406102_WPSOffice_Level1)**

[2.1.开发工具](#_Toc860615211_WPSOffice_Level2) [4](#_Toc860615211_WPSOffice_Level2)

[2.2. JavaScript](#_Toc976096520_WPSOffice_Level2) [4](#_Toc976096520_WPSOffice_Level2)

[2.3.Python](#_Toc2117603645_WPSOffice_Level2) [4](#_Toc2117603645_WPSOffice_Level2)

[2.4.Django框架](#_Toc794363179_WPSOffice_Level2) [4](#_Toc794363179_WPSOffice_Level2)

[2.5.Vue框架](#_Toc145096724_WPSOffice_Level2) [5](#_Toc145096724_WPSOffice_Level2)

[2.6.MariaDB数据库](#_Toc1135596424_WPSOffice_Level2) [5](#_Toc1135596424_WPSOffice_Level2)

[2.7.搭建env虚拟环境](#_Toc1350449956_WPSOffice_Level2) [5](#_Toc1350449956_WPSOffice_Level2)

[2.8.前后端分离](#_Toc1716359548_WPSOffice_Level2) [5](#_Toc1716359548_WPSOffice_Level2)

**[第三章 构架概述](#_Toc860615211_WPSOffice_Level1)** **[7](#_Toc860615211_WPSOffice_Level1)**

[3.1.系统架构图](#_Toc339619014_WPSOffice_Level2) [7](#_Toc339619014_WPSOffice_Level2)

[3.2.架构概述](#_Toc728856266_WPSOffice_Level2) [7](#_Toc728856266_WPSOffice_Level2)

[3.3.学生选课流程图](#_Toc393617786_WPSOffice_Level2) [7](#_Toc393617786_WPSOffice_Level2)

[3.4.系统模型E-R图](#_Toc1256967935_WPSOffice_Level2) [8](#_Toc1256967935_WPSOffice_Level2)

[3.5.类的设计](#_Toc594733852_WPSOffice_Level2) [9](#_Toc594733852_WPSOffice_Level2)

**[第四章 前端开发](#_Toc976096520_WPSOffice_Level1)** **[10](#_Toc976096520_WPSOffice_Level1)**

[4.1. Ajax技术：](#_Toc414225480_WPSOffice_Level2) [10](#_Toc414225480_WPSOffice_Level2)

[4.2.HTML5技术：](#_Toc573090864_WPSOffice_Level2) [10](#_Toc573090864_WPSOffice_Level2)

[4.3.CSS技术：](#_Toc1728147514_WPSOffice_Level2) [11](#_Toc1728147514_WPSOffice_Level2)

[4.4.B/S结构：](#_Toc824545949_WPSOffice_Level2) [11](#_Toc824545949_WPSOffice_Level2)

[4.5.浏览器连接限制：](#_Toc877106528_WPSOffice_Level2) [11](#_Toc877106528_WPSOffice_Level2)

**[第五章 系统测试](#_Toc2117603645_WPSOffice_Level1)** **[12](#_Toc2117603645_WPSOffice_Level1)**

[5.1.测试目的：](#_Toc1230436247_WPSOffice_Level2) [12](#_Toc1230436247_WPSOffice_Level2)

[5.2.测试方案：](#_Toc1751992140_WPSOffice_Level2) [12](#_Toc1751992140_WPSOffice_Level2)

[5.3.测试目标：](#_Toc2013238241_WPSOffice_Level2) [13](#_Toc2013238241_WPSOffice_Level2)

[5.4.测试参考文档：](#_Toc1387413579_WPSOffice_Level2) [13](#_Toc1387413579_WPSOffice_Level2)

[5.5.术语和缩略语：](#_Toc773490637_WPSOffice_Level2) [13](#_Toc773490637_WPSOffice_Level2)

[5.6.单元测试：](#_Toc967147874_WPSOffice_Level2) [14](#_Toc967147874_WPSOffice_Level2)

[5.7.集成测试：](#_Toc796551428_WPSOffice_Level2) [15](#_Toc796551428_WPSOffice_Level2)

[5.8.测试用例：](#_Toc1807569530_WPSOffice_Level2) [16](#_Toc1807569530_WPSOffice_Level2)

[5.9.关注点——文本输入框](#_Toc1275790141_WPSOffice_Level2) [17](#_Toc1275790141_WPSOffice_Level2)

[5.10.关注点——下拉列表](#_Toc622335276_WPSOffice_Level2) [18](#_Toc622335276_WPSOffice_Level2)

[5.11.关注点——增加数据](#_Toc1758128041_WPSOffice_Level2) [18](#_Toc1758128041_WPSOffice_Level2)

[5.12.关注点——修改数据](#_Toc1667727453_WPSOffice_Level2) [18](#_Toc1667727453_WPSOffice_Level2)

[5.13.关注点——删除数据](#_Toc104257730_WPSOffice_Level2) [18](#_Toc104257730_WPSOffice_Level2)

[5.14.关注点——查询数据](#_Toc471259604_WPSOffice_Level2) [19](#_Toc471259604_WPSOffice_Level2)

[5.15.关注点——数据接入与处理](#_Toc496340325_WPSOffice_Level2) [19](#_Toc496340325_WPSOffice_Level2)

[5.16.其他](#_Toc74377727_WPSOffice_Level2) [19](#_Toc74377727_WPSOffice_Level2)

**[第六章 系统更新](#_Toc794363179_WPSOffice_Level1)** **[19](#_Toc794363179_WPSOffice_Level1)**

[6.1.系统不足](#_Toc1265622784_WPSOffice_Level2) [19](#_Toc1265622784_WPSOffice_Level2)

**[第七章 总结](#_Toc145096724_WPSOffice_Level1)** **[19](#_Toc145096724_WPSOffice_Level1)**

[7.1.总结](#_Toc641437049_WPSOffice_Level2) [19](#_Toc641437049_WPSOffice_Level2)

1. **绪论**

# **目的及意义**

此系统是根据之前的0.2版本改进升级而得来的，在选课这方面有了很大的进步，同时还增加教师评分模块，尽管显得有些多余，系统依然从以下三个方面考虑：

管理者的角度：对于一个管理者来说，不论是管理还是维护一个系统，都希望能够通过简单的操作来完成复杂的工作。考虑至此，我们专门为管理员在系统中，设置了一个系统设置的模块，在此模块中，管理员可以看到系统的绝大部分关键部分，极大地方便了管理员管理和维护系统。

教师的角度：对于日常工作繁多的老师们来说，她们最需要的就是知道一张时刻表，让他们了解到自己在什么时候要去上课，方便她们非上课时间安排其他工作事宜。因此，本系统中专门设置了一个课程查询模块，教师只需在教师课程查询模块输入自己的工号，即可进入系统看到自己的课程安排表。

学生的角度：学生的需求很简单，只需要满足他们选课的要求，在选完课之后给他们一张属于自己的课表即可。本系统专门为学生设置了一个学生网上选课模块，方便学生选择自己喜欢的课程和老师。然后在课程查询模块有一个学生课程查询功能，学生只需输入自己的学号，即可得到与自己所选课程相对应的课程表。

1. **开发工具及关键技术介绍**

## 2.1.开发工具

本学生选课系统开发用到的开发工具主要为VSCode，是一款免费开源且跨平台的现代化轻量级代码编辑器，几乎支持所有主流开发语言。

**2.2. JavaScript**

JavaScript是一种高级的、多范式、解释型的编程语言，是一门基于原型、函数先行的语言，它支持面向对象编程、命令式编程以及函数式编程。

**2.3.Python**

Python是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。

**2.4.Django框架**

Django框架是一个开放源代码的Web应用框架，由Python写成。采用MTV的框架模式，即模型M，视图V，和模板T。

**2.5.Vue框架**

Vue是一套用户构建用户界面的渐进式框架。与其他大型框架不同，Vue被设计为可以自底向上逐层应用。Vue的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种类库结合使用时，Vue也完全能够为复杂的单页面应用提供驱动。

**2.6.MariaDB数据库**

MariaDB数据库管理系统是MySQL的一个分支，主要由开源社区在维护，采用GPL授权许可 MariaDB的目的是完全兼容MySQL，包括API和命令行，使之能轻松成为MySQL的代替品。在存储引擎方面，使用XtraDB（英语：XtraDB）来代替MySQL的InnoDB。

MariaDB是基于事务的Maria存储引擎，替换了MySQL的MyISAM存储引擎，它使用了Percona的 XtraDB，InnoDB的变体。

**2.7.搭建env虚拟环境**

在macOS/Linux系统下：

sudo apt-get install python3-venv

python3 -m venv env

**2.8.前后端分离**

前后端不分离：

在前后端不分离的应用模式中，前端页面看到的效果都是由后端控制，由后端渲染页面或重定向，也就是后端需要控制前端的展示，前端与后端的耦合度很高。

前后端分离：

前后端分离的应用模式中，后端仅返回前端所需的数据，不再渲染HTML页面，不再控制前端的效果。至于前端用户看到什么效果，从后端请求的数据展现都是后端通过异步接口(AJAX/JSONP)的方式提供的，前端只管展现。

前后端分离的优点：

1.彻底解放前端

前端不再需要向后台提供模板或是后台在前端html中嵌入后台代码

2.提高工作效率，分工更加明确

前后端分离的工作流程可以使前端只关注前端的事，后台只关心后台的活，两者开发可以同时进行，在后台还没有时间提供接口的时候，前端可以先将数据写死或者调用本地的json文件即可，页面的增加和路由的修改也不必再去麻烦后台，开发更加灵活。

3.局部性能提升

通过前端路由的配置，我们可以实现页面的按需加载，无需一开始加载首页便加载网站的所有的资源，服务器也不再需要解析前端页面，在页面交互及用户体验上有所提升。

4.降低维护成本

通过MVC框架，我们可以非常快速的定位及发现问题的所在，客户端的问题不再需要后台人员参与及调试，代码重构及可维护性增强。

鉴于前后端分离的诸多优点，我们在系统设计中采用了前后端分离技术，减轻后端负担，将界面渲染完全交于前端，同时也可以大大降低维护成本和维护难度。

1. **构架概述**

**3.1.系统架构图**

|  |
| --- |
| /home/eternity/day10/架构图.png架构图 |
| 图3.1 学生选课系统架构图 |

**3.2.架构概述**

此学生选课系统，主要由系统设置、教学安排、网上选课、评分五大模块组成，其中专业、课程、教师、学生以及教学计划的设计，都是为了达到最终选课的目的，而课程查询则是对选课之后结果的补充 。

**3.3.学生选课流程图**

|  |
| --- |
| **流程图** |
| 图3.2学生选课流程图 |

**3.4.系统模型E-R图**

|  |
| --- |
| **/home/eternity/day10/E-R图.pngE-R图** |
| 图3.3学生选课系统E-R图 |

**3.5.类的设计**

学生选课系统中的类较多且复杂，各类之间的关系如图所示：

|  |
| --- |
| **/home/eternity/day10/类图.png类图** |
| 图3.4学生选课系统类图 |

1. **前端开发**

**4.1. Ajax技术：**

Ajax是一种创建交互式网页应用的网站前端开发技术，其核心是JavaScript对象——XmlHttpRequest。Ajax主要提供与服务器异步通信的能力，从而使用户从请求/响应的循环中解脱出来。借助于Ajax可以在用户单击按钮时使用JavaScript和DHTML立即更新UI，并向服务器发出异步请求，以执行更新或查询数据库的操作。

**4.2.HTML5技术：**

HTML5是构建Web内容的一种语言描述方式。HTML5是Web中核心语言HTML的规范，用户使用任何手段进行网页浏览时看到的内容原本都是HTML格式的，在浏览器中通过一些技术处理将其转换成为了可识别的信息。

**4.3.CSS技术：**

层叠样式表(英文全称：Cascading Style Sheets)是一种用来表现HTML（标准通用标记语言的一个应用）或XML（标准通用标记语言的一个子集）等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。

**4.4.B/S结构：**

在浏览网页的过程中，用户端向服务器发出访问请求，到服务器反馈、网页内容重现需要经历以下步骤：输出网页URL→通过域名解析IP地址→向Web服务器发送访问请求→服务器确认请求寻找资源文件→服务器反馈→服务器发送HTML文件→浏览器接收HTML文件并请求资源文件→服务器确认资源文件发送给浏览器→浏览器接收资源并重新渲染→访问完成。B/S结构直至今日，仍旧是Web网页访问过程的基础。

**4.5.浏览器连接限制：**

为避免服务器堵塞，在HTTP协议中有着详尽的规定，服务器、客户端之间的并发连接数有着明确的限制，在同一个时间点，服务器和用户端之间的持久性连接数为1个。但是在实际访问Web网页的过程中，却难以避免出现二次连接问题。客户端在访问Web网页的过程中触发了多次Ajax或者用户在访问网页的过程因误操作发送了两个访问Web的请求，如上文所述，从用户发送请求到Web网页内容呈现必定会经过一段时间，而用户在Web网页没有反馈的情况下，又发送了一次Web访问请求，这个时候就会出现二次连接问题。

1. **系统测试**

**5.1.测试目的：**

为系统开发项目管理者、软件工程师、系统维护工程师、测试工程师提供关于项目系统整体功能和性能的测试指导，同时也是用户确定软件是否完整测试的重要依据。

**5.2.测试方案：**

系统的功能测试选用了手工测试，运用黑盒测试中的等价类划分、边界值分析、错误推测、因果图法。系统UI方面的测试包括：易用性测试、规范性测试、帮助设施测试、合理性测试、美观与协调性测试、独特性测试、快捷方式组合测试。系统的安全性、兼容性、安装与反安装、配置测试也是手工测试。单元测试采用的方法是白盒测试，功能测试采用黑盒测试。

|  |
| --- |
| /home/eternity/图片/2020-08-28 21-25-20 的屏幕截图.png2020-08-28 21-25-20 的屏幕截图 |
| 图5.1系统测试方案架构图 |

**5.3.测试目标：**

在用户使用软件之前，尽可能的发现软件中存在的错误和不合理之处，排除软件中潜在的错误，最终把高质量的软件系统交付给用户。系统的测试覆盖范围：功能、性能、UI、安全性、兼容性、容量。

**5.4.测试参考文档：**

GBT 15532-2008 计算机软件测试规范  
GBT 9386-2008 计算机软件测试文档编制规范

**5.5.术语和缩略语：**

本文使用的术语、定义

|  |  |
| --- | --- |
| 软件测试 | 软件测试就是利用测试工具按照测试方案和流程对产品进行功能和性能测试，甚至根据需要编写不同的测试工具，设计和维护测试系统，对测试方案可能出现的问题进行分析和评估。 |
| 系统测试 | 系统测试是通过与系统的需求规格作比较，发现系统与系统需求规格不相符合或与之矛盾的地方。它将通过确认测试的系统，作为整个基于计算机系统的一个元素，与计算机硬件，外设，某些支持软件，数据和人眼等其他系统元素结合起来，在实际使用环境下，对计算机系统的测试。 |
| 功能测试 | 黑盒测试是基于系统需求规格，在不知系统或组件的内部结构的情况下进行测试，通常又将黑盒测试叫做基于规格的测试，输入输出测试，功能测试或数据驱动测试，是基于用户观点出发的测试。主要是验证功能是否符合要求，包括原定功能的校验，是否有冗余功能，遗漏功能。 |
| 性能测试 | 性能测试关注的系统的整体，它和通常说的强度测试、压力测试、负载测试有密切关系。所以压力测试和强度测试应与性能测试一同进行。 |

**5.6.单元测试：**

首先依照系统、子系统和模块进行划分，但最终的单元必须是功能模块，或面向对象过程中的若干个类，单元测试是对功能模块进行正确性检验的测试工作，也是后续测试的基础。目的在于发现各模块内部可能存在的各种差错，因此需要从程序的内部结构出发设计测试用例，着重考虑以下五个方面：

（1）模块接口：对所测模块的数据流进行测试；

（2）局部数据结构：检查不正确或不一致的数据类型说明、使用尚未赋值或尚未初始化的变量、错误的初始值或缺省值。

（3）路径：虽然不可能做到穷举测试，但要设计测试用例查找由于不正确的计算（包括算法错、表达式的符号表示不正确、运算精度不够等）、不正确的比较或不正常的控制流（包括不同数据类型量的相互比较、不适当地修改了循环变量、错误的或不可能的循环终止条件等）而导致的错误。

（4）错误处理：检查模块有没有对于常见错误的条件设计比较完善的错误处理功能，保证其逻辑上的正确性。

（5）边界：注意设计数据流、控制流中刚好等于、大于或小于确定的比较值的用例。

**5.7.集成测试：**

集成测试也叫组装测试或联合测试。通常，在单元测试的基础上需要将所有的模块按照设计要求组装成系统，这时需要考虑的问题如下：

（1）把各个模块连接起来，模块接口的数据是否会丢失；

（2）一个模块的功能是否会对另一个模块的功能产生不利的影响；

（3）各个子功能组合起来，能否达到预期要求的父功能；

（4）全局数据结构是否有问题；

（5）单元模块的误差累积起来，是否会放大，从而达到不能接受的程度。我们在组装时可参考采用一次性组装方式或增值式组装方式；

**5.8.测试用例：**

注：由于学生选课系统中，功能复杂，模块繁多，以下只取部分测试用例作为参考。

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例ID | 001 | | |
| 测试用例名称 | 非法用户登录选课界面 | | |
| 产品名称 | 学生选课系统 | 产品版本 | 0.2 |
| 功能模块名 | 学生选课 | 测试平台 |  |
| 用例入库者 | Eternity | 用例更新者 | Eternity |
| 用例入库时间 | 2020-8-29 | 用例更新时间 | 2020-8-29 |
| 测试功能点 | 输入错误的学号 | | |
| 测试目的 | 阻止非法用户登录系统 | | |
| 测试级别 | 高 | | |
| 预置条件 | 登录学号为系统中存在的 | | |
| 测试步骤 | 1. 输入系统中不存在的学号 2. 点击登录 | | |
| 预期结果 | 1. 系统拒绝用户登入 2. 提示学号不存在，请输入正确的学号 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例ID | 002 | | |
| 测试用例名称 | 编辑中取消返回原数据 | | |
| 产品名称 | 学生选课系统 | 产品版本 | 0.2 |
| 功能模块名 | 系统设置 | 测试平台 |  |
| 用例入库者 | Eternity | 用例更新者 | Eternity |
| 用例入库时间 | 2020-8-29 | 用例更新时间 | 2020-8-29 |
| 测试功能点 | 在编辑中点击取消返回原值 | | |
| 测试目的 | 测试数据与数据库间的联系 | | |
| 测试级别 | 中 | | |
| 预置条件 | 编辑数据格式合法 | | |
| 测试步骤 | 1.任意选中编辑一栏数据  2.编辑中途点击取消 | | |
| 预期结果 | 表格中返回原数据 | | |

**5.9.关注点——文本输入框**

（1）检查空数据；

（2）检查过长数据（超出空间本身的长度和数据库中改字段所允许的长度）；

（3）检查特殊字符，尤其是数据库中不允许的字符，甚至回车字符、空格字符等；

（4）检查字符类型，比如应该输入数字的文本框输入英文字符；

（5）中文字符的处理；

（6）对于日期时间型数据，检查格式正确性以及时间日期的合理性。比如开始时间不能晚于结束时间等；

**5.10.关注点——下拉列表**

（1）列表数据是否正确、完整；

（2）下拉列表与其他空间的联动关系；

（3）是否允许多选；

**5.11.关注点——增加数据**

（1）数据个数的上限；

（2）重复数据处理，尤其是键值的重复；

（3）相关表格的更新；

（4）检查多次使用back键的情况，在有back的地方，back回到原页面，再back重复多次，看是否会出错；

**5.12.关注点——修改数据**

（1）不能破坏数据库数据的关联和完整；

（2）重复数据处理，尤其是键值的重复；

（3）修改登录用户本身信息时对系统的影响；

（4）修改正在使用的数据；

（5）检查多次使用取消键的情况，在有取消的地方，取消，回到原页面，再取消，重复多次，看是否会出错。

**5.13.关注点——删除数据**

（1）不能破坏数据库数据的关联和完整；

（2）删除正在使用的数据；

（3）删除登录用户本身；

**5.14.关注点——查询数据**

（1）多条件组合查询的正确性；

（2）多次连续查询正确性；

**5.15.关注点——数据接入与处理**

（1）数据接入方式是否全部可用，数据是否能正确接入；

（2）数据处理方式是否全部可行；

（3）数据的动态监测是否正确无误；

**5.16.其他**

（1）对网络故障的提示；

（2）同一用户多次登录；

（3）内存使用情况；

（4）压力测试，系统承受能力，多用户同时登录使用。

1. **系统更新**

**6.1.系统不足**

系统中功能并未完善，例如学生在选择课程时，只能选择一门课程，主要原因是因为在建立模型之时，选课清单与学生之间的一对一关系。从而导致学生在查询课程只能查询到一节课，并未实现真正意义上的选课。

1. **总结**

**7.1.总结**

本系统是笔者通过Django后端框架，以及Vue前端框架整合起来的简单系统，同时还引用element-ui、axios等组件。首先系统设计过程中采用了前后端分离技术，大大降低了维护难度和成本。其次系统在设计过程中考虑的最多的就是用户体验，尽最大努力降低系统操作难度，因此操作简单也是本系统的一大亮点。当然，系统除了以上优点还存在以下不足之处，第一点，系统安全性严重不足，系统框架代码显而易见，API调用简单明了，并没有做任何修饰，这是系统最大的致命点；第二点，在0.2版本的学生选课模块，相较0.1版本的选课模块，学生选课模块不仅没有达到预期要求，相较之下功能在0.1版本的基础之下有明显退步。最后，相比0.1,版本，0.2版本有进步，但也有缺陷，整体看来退步大于进步，所以说0.2版本是一个失败的版本，这其中的主要原因主要是系统模型设计上出现了较大错误。系统的亮点我们会在0.3版本继续保持并持续优化达到更好的效果，系统的不足将会在下一个版本进行改进。