

编号: _____



桂林电子科技大学
GUILIN UNIVERSITY OF ELECTRONIC TECHNOLOGY

毕业设计说明书

题 目: 基于 HTML5 的培训机构
的排课系统的设计与实现

学 院: 计算机与信息安全学院

专 业: 软件工程

学生姓名: 梁艺可

学 号: 1600300522

指导教师: 潘海玉

职 称: 副教授

题目类型: ☐理论研究 ☐实验研究 ☐工程设计 ☐工程技术研究 ☒软件开发

2020 年 06 月 07 日

摘 要

新中国成立以后，人民生活水平等各方面的质量都有了很大的改善。加之教育一直以来都是被中国人民所注重的，因此也越来越多的人接受更高的教育，越来越多的人进入教育行业，教学课程日益多样化；各方面资源数量的快速增长，让传统的手工排课再也不太适用于当下的教育环境，即使可以通过人工去排课也会导致这其中付出很多的精力，工作量巨大，况且很多时候还需要考虑个别教师的上课喜好，尽量人性化。

为了解决人工排课的困难，本文通过对人工智能领域的遗传算法进行研究分析，开发出了一套基于该算法的培训机构的排课系统，该系统针对于高中阶段教育的培训机构，学校教务处方面首先需要给某一个需要开设课程的学期分配好课程计划，即：年级编号，班级编号，教师编号，课程编号，上课周数，每周课时数，课程是否固定上课时间，课程属性等基本信息；在应用到算法实现上的时候，即可将课程计划的各项信息，即以上的信息编号进行编码，编码规则为：是否固定上课时间+年级编号+班级编号+教师编号+课程编号+课程属性+上课时间进行编码，即字符串的拼接操作，在完成编码以后即可得到初始的基因编码，这些基因编码的集合，我们将其看作初始种群，每一条基因编码就看作是种群中的一个个体，随后将每一个初始基因编码为其随机分配一个时间片，再将这些个体经过不断地进行交叉、选择、变异操作，交叉、变异都是编码中的时间进行操作，选择则是评估交叉后的种群适应度值，值大的则进入下一代，到达指定遗传代数后算法结束，随后开始给这些经过变异操作的基因编码分配教室，判断“时间-教室-讲师”条件(即同一时间同一教室只能上一门课，一个讲师同一时间只能在一个教室上一门课)不冲突最终即可得到每一门课程的上课时间以及上课所在的教室。

系统采用前后端分离的模式进行开发，技术实现上，使用了当前最为流行的 Spring 框架进行开发，前端使用了 Vue 框架整合 Element UI 组件实现；后端使用 SpringBoot 框架搭建，编程语言使用 Java 1.8，数据库为 MySQL 8.x 版本；目前，很多排课软件都是采用了 C/S 架构(即客户端/服务器)进行开发，这样的话只有安装了其客户端才可以使用；本次为了用户更方便采用了 B/S 架构(即浏览器/服务器)，用户只需要一个浏览器即可使用该系统，带来了极大的便利。

关键词：遗传算法；排课；教育；基因；个体；种群；编码；培训机构

Abstract

Since the founding of New China, the quality of people's living standards and other aspects have greatly improved. In addition, education has always been valued by the Chinese people, so more and more people are receiving education, more and more people are entering the education industry, and teaching courses are becoming more and more diverse. The rapid increase in the number of resources in all aspects makes traditional manual courses less suitable for the current educational environment. Even if it is possible to arrange lessons manually, it will result in a lot of energy and huge workload; moreover, it is often necessary to consider the preferences of individual teachers in class and implement humanized teaching as much as possible.

In order to solve the difficulty of manual scheduling, this paper has developed a set of scheduling system for training institutions based on the algorithm through research and analysis of genetic algorithms in the field of artificial intelligence. This system is aimed at training institutions for high school education. The specific method is that the school's Academic Affairs Office first needs to assign a course plan to a semester that needs to open a course, that is: grade number, class number, teacher number, course number, class week, class hours per week, and whether the course is fixed. Basic information such as course attributes; when applied to the implementation of the algorithm, you can encode the information number of the information of the above course plan; the coding rule is: whether the class time is fixed + grade number + class number + teacher number + Course number + Course attribute + Class time encoding, that is, string concatenation operation, you can get the initial gene code after completing the coding; these gene code collection, we regard it as the initial population, each gene code is Seen as an individual in the population, then each initial gene encoding was randomly assigned to a time slice, and then these individuals were continuously subjected to crossover and mutation operations. These two steps are operations on time encoding to reach the designated The algorithm ends after genetic algebra. Afterwards, we began to assign classrooms to these genetically modified genetic codes to judge the "time-classroom" conditions (that is, only one course can be taken in the same classroom at the same time) without conflict. Eventually, we can get the class time of each course and the class classroom.

The system is developed using a separate front-end and back-end mode. For technical implementation, the most popular Spring framework is used for development. The front-end uses the Vue framework and Element UI components for implementation. the back end is built using the SpringBoot framework, the programming language is Java, the version is JDK1.8, and the database is MySQL 8.x version; at present, many scheduling software is using C / S

architecture (ie client / server). Development, in this case, only the client can be used; this time, for the convenience of users, the B / S architecture (that is, browser / server) is adopted. The user only needs a browser to use the system, which gives the user Brought great convenience.

Key words: genetic algorithm; class scheduling; education; gene; individual; population; coding; training institution

目 录

1 绪论	1
1.1 选题背景.....	1
1.2 发展状况.....	1
1.3 行业前景.....	2
1.4 研究意义.....	3
1.5 本文组织结构.....	3
2 可行性分析.....	3
2.1 经济可行性.....	3
2.2 社会可行性.....	4
2.3 技术可行性.....	4
3 需求分析	5
3.1 任务概述.....	5
3.1.1 目标.....	5
3.1.2 假定和约束.....	5
3.1.3 用户特点.....	6
3.2 需求规定.....	6
3.2.1 对功能的规定	6
3.2.2 对非功能的规定	7
3.2.3 对性能的规定	7
3.3 系统用例图.....	8
3.3.1 学生用例图	8
3.3.2 管理员用例图	9
3.3.3 讲师用例图	10
3.4 运行设备规定	10
3.4.1 设备.....	10
3.4.2 支持的软件.....	10
3.4.3 控制.....	10
4 总体设计	11
4.1 系统层次方框图	11
4.2 接口设计.....	11
4.2.1 学生接口	11
4.2.2 管理员接口	12
4.2.3 讲师接口	12
4.2.4 开课任务接口	12

4.2.5 教材信息接口	13
4.2.6 班级接口	13
4.2.7 教室接口	13
4.2.8 上课计划接口	13
4.2.9 文档接口	14
4.2.10 练习题类型接口	14
4.2.11 练习题接口	14
4.2.12 教学区域接口	14
4.2.13 教学楼接口	15
4.2.14 云存储接口	15
4.2.15 系统数据接口	15
4.3 数据库设计	15
4.3.1 用户关系 E-R 图	15
4.3.2 教学设施与年级班级学生关系 E-R 图	16
4.3.3 排课任务上课任务与课程关系 E-R 图	17
4.3.4 习题类别与习题 E-R 模型图	17
4.3.5 其它 E-R 图	17
4.3.6 数据表设计	18
5 详细设计	24
5.1 注册模块	24
5.1.1 学生注册	24
5.1.2 讲师注册	25
5.2 登录模块	25
5.2.1 用户登录	25
5.3 修改用户信息	26
5.3.1 修改信息	26
5.3.2 修改密码	27
5.4 课程管理	28
5.4.1 网课管理	28
5.4.2 教材管理	28
5.5 排课模块	29
5.5.1 下载排课任务 Excel 模板	29
5.5.2 Excel 导入排课任务	30
5.5.3 手动添加排课任务	30
5.5.4 排课	31

5.5.5 查看课表.....	32
5.6 学生管理.....	32
5.7 讲师管理.....	33
5.8 教学资料管理.....	33
5.8.1 在线测试题库管理	33
5.8.2 作业学习文档管理	34
5.9 班级管理.....	35
5.9.1 查看班级.....	35
5.9.2 添加班级.....	35
5.10 教学设施管理	35
5.10.1 教学楼管理	36
5.10.2 教室管理.....	36
5.10.3 教学区域管理	37
5.11 学生端功能	37
6 系统实现	38
6.1 注册模块.....	38
6.1.1 学生注册.....	38
6.1.2 讲师注册.....	38
6.2 登录模块.....	39
6.3 个人信息修改模块	40
6.4 系统数据可视化	41
6.5 排课管理模块	41
6.5.1 排课任务列表	41
6.5.2 Excel 导入任务	42
6.5.3 手动添加任务	42
6.5.4 排课实现.....	42
6.5.5 查看课表.....	44
6.6 课程管理.....	45
6.6.1 教材管理.....	45
6.6.2 网课类别管理	46
6.6.3 网课管理.....	46
6.7 讲师管理.....	48
6.8 班级管理.....	48
6.9 学生管理.....	49
6.10 教学资料管理	50

6.10.1 题库管理	50
6.10.2 学习文档	51
6.11 教学设施管理	52
6.11.1 教学楼管理	52
6.11.2 教室管理	53
6.11.3 教学区域管理	53
7 系统测试	54
7.1 学生注册测试	54
7.2 登录测试	55
7.3 排课测试	56
7.4 导入排课任务测试	57
8 结论与展望	57
8.1 结论	57
8.2 展望	58
谢 辞	59
参考文献	60

1 绪论

1.1 选题背景

20 世纪中后期到现在的 21 世纪，我国在各方面的成绩突飞猛进，尤其是在经济发展水平和科学技术上，取得了世界瞩目的成就，这也直接使得国民生活水平质量得到了巨大的提升，同时中国社会人才短缺的问题也逐渐暴露出来，对人才的需求迫在眉睫，国民对教育的重要性有了新的认识，重视程度有了新的提升，而这只有加大对教育的投资，加强对教育的重视才可以弥补职业教育不足带来的人才短缺问题。加之 20 世纪中后期我国“科教兴国”，“人才强国”战略的提出，更是凸显出了国家对于教育的重视程度以及人才需求的迫切性，因而在这之后，国家以及社会各界对教育行业的投入持续增长，这也随之促使了教育教学等资源的极速增长。而对于国民来说，一方面，意识形态的改变，让许多家长非常重视其子女的教育，甚至家长本身也对自己知识水平有了更高的要求，另一方面，社会发展需要更多的人才，“人才强国”战略也督促着人们想要在社会上有更好的发展，能够有更好的生活，也必须学习，必须接受教育，毕竟惯性思维里“知识改变命运”一直是不变的道理。基于以上各种因素，我国从那之后接受教育的人数、群体都呈现出只增不减的状态，国家加大对教育的投资，对学生的各种有利政策，学校扩招，除了公立学校之外，越来越多的人也参与到了教育行业中，为人才建设贡献一份力量，例如许多私立学校的创办，培训机构的设立都是教育行业蒸蒸日上的真实写照，教育教学资源的多元化，教育机构的大规模化，也寓意着在教育教学开展中也增添了更多的元素。此外，家长们为了提高孩子们的成绩，也更多地让自己的子女参加课外的一些培训课程，在这样的状况下，开设的课程必然是越来越多的，由此而来，教学安排中的排课事项肯定是不能像传统那样子通过人工手动去排课的，人工手动排课必将花费大力的人力物力；如今科学技术如此发达，计算机发展也相当可观，硬件成本也不像以前那么昂贵，计算机的处理能力、运算能力空前强大，我们应该合理地利用计算机去解决生产中的问题，本文我们将以遗传算法为基础，通过计算机编程解决排课问题，利用程序将大力减少了人力物力的损耗，提高工作效率。

1.2 发展状况

上个世纪 50 年代末人们就已经开始着手对课程编排问题进行了研究并尝试在计算机上解决该问题，针对该问题的解决方案，他们也曾解决排课问题的数学模型，是否存在对排课问题的解等方面上进行了许多的研究，但都没有很好的解决方案；直到上个世纪 70 年代，1976 年，美国人 S.Even 等对排课问题的时间复杂度和空间复杂度进行了分析和研究，证明排课问题是一个 NP 完全问题^[1]也就是说算法的运算时间是呈现出指

数增长的样式的，在数学领域是没有一个通用的算法能够很好地解决该问题的，从此人们才开始将视线从数学模型的研究转移到解决该问题的实用算法探索上。

在最开始，人们就是通过在黑板上拿着笔去填写表格，边填写格子边观看上面的课程安排，不合理的地方就擦掉重新安排的，直到设定的课程都能够安排上为止，这要求编排者需要有非常严谨的思维能力，这样子的方式也只能适用于待安排的课程比较少的环境下，才可以在不耗费大功夫的情况下通过人工去手动编排，否则将会给人带来巨大的困难，导致人力物力等成本的增加，降低工作效率。

到了 20 世纪 80 年代，随着计算机得到发展以及普及，许多人也能运用上这一拥有强大计算能力的机器，这个时候，人们在此之前已经对算法有了许多的了解，例如常用与解决问题的贪心算法、分治算法、动态规划法；因而人们这时候也想到了尝试通过使用计算机编程算法去解决课程安排问题，以让教育工作者提升工作效率，这个阶段，人们通常采用的是贪心算法、分治算法以及回溯算法进行处理排课问题完成排课工作。这几种常用的算法也在排课实现中有着不错的表现，但是它们也有着自己的缺点，例如贪心算法虽然没有迭代，时间复杂度相对较低，但是其贪心策略也常常因为课程任务过多导致计算得到的是局部最优解而非全局最优解；回溯算法时间复杂度会随着问题规模的增长而快速增大，因为该算法需要不断地进行迭代，这种情况下可能会导致栈溢出。

到了近些年，科技进一步发展，计算机领域的发展更是惊人，人工智能这一名词也常常闻于耳，许多算法也在人工智能领域被深度学习等方式进一步优化，例如本次我们使用的遗传算法，该算法在 20 世纪 70 年代就已经被提出，遗传算法是根据达尔文的进化论得来，即是模拟生物进化，适者生存，不适者淘汰的自然规律的一种全局搜索最优解算法，原理就是利用数学的方式，通过计算机去仿真模拟生物在进化过程中出现的选择、交叉、变异再到淘汰的过程，不断地进行代数的遗传，达到终止条件时算法就会结束。其在解决组合优化问题上表现出了优异的成绩，因而也一直被人们所喜爱，发展至今，遗传算法已经可以很好地处理课程安排问题，我们这次也毅然决定使用该算法进行尝试实现排课系统。

1.3 行业前景

之前我们所述，已经知道了随着国家综合实力的增强，重视教育已经促使各式各样的教育机构遍地开花，许多研究数据也表明，直到 2019 年排课软件所占的市场份额已经超过了亿元级别大关，从这些数据中我们应该也可以看到，在未来教育行业更加蓬勃发展的时代，人们唯有通过排课软件去提升课程安排工作的效率，因而排课软件在未来的市场份额不但一点也不比现在差，反而可能会出现比较大幅度的增长趋势。

1.4 研究意义

通过对排课系统的研究、设计与实现，利用计算机去解决传统中手工完成的工作，将传统通过人工手动去处理的计算问题交由每秒钟计算次数以亿为单位计量的处理器去处理，这无疑在很大程度上提升了教育工作者的工作效率，摆脱了传统手工操作给人带来的束缚，提升了用户体验也促进了行业的发展。

1.5 本文组织结构

本文先通过摘要对文章总体做一个概括，摘要之后有相关的关键词，让读者对文章的描述内容以及条理也比较清晰，从中也知道了本文是关于排课问题的研究；随后通过引言引出了本文的相关背景，通过背景的描述，教育教学资源的增加，科技的进步，让我们有了更好的办法去解决排课问题而不是停留在传统的人工去实现排课，从而得到本文的编写的缘由；再而对排课研究的发展状况进行了一个简要的描述，从而了解到前人在排课问题上所作的努力及其取得的成果，再而道出了本文所进行的研究的意义。

接下来开始分析本文所研究的排课系统在经济、政策、技术上的可行性；在可行性通过之后开始对系统进行一个比较常规的需求分析，这也是软件开发中所必须经历的流程；在明确我们的需求之后，那么就大体上知道我们要做什么了，那么怎么做呢？那么先来个大概要怎么做，此时，我们就需要进入到了总体设计，从而整体上有了实现的思路，接下来开始对系统进行详细的设计；设计思路明确后，还需要编码实现，因而接下来需要做系统实现，这也是排课系统中比较关键的部分；系统实现了之后，需要对系统的功能等各方面进行测试，最起码的就是对功能进行测试，以确保系统的可用性。最后得到本文研究的结论；致谢，向指导老师以及参考资料的作者表示感谢。

2 可行性分析

2.1 经济可行性

前期分析，针对排课算法实现排课系统的这一需求，一方面，在软件开发过程中所需成本不高，目前作为个人开发者，只需要拥有自己的计算机即可，在计算机上安装好所需的开发环境，另外计算机本身在开发阶段也可以作为部署项目的服务器，如果后续需要使用到专业的服务器，针对于该排课系统，并发量极少，因而说在服务器费用支出部分也不会需要过多的费用，因为租赁相对较为便宜的服务器即可运行该系统并完成常规下的任务。另一方面，前期系统由单人进行开发，不需要额外请人导致雇佣费用的支出，在人力成本上也并非很高，都是在可以接受的范围内有序进行开发。

后期分析如果后续系统涉及到商业应用，那么对于此情况下，整体上来说，需要对

该系统进行定制化的改造，因为通常来说，针对不同的教育机构，不同的学校，其教学生态体系都是有着不少的区别的，因而针对他们的模式也常常需要进行定制，或者说在系统的原有基础上进行功能的扩展，使得系统可以应对多种生态的教学模式，针对大部分的情况都可以进行排课，满足学校的课程安排问题，如果在这方面做得到，推广得好，使用效果，实用性，可靠性等方面都基本上可以满足学校的需求，我相信这套软件系统也可以依靠其可靠性取得比较良好的经济效益的。

综合以上分析，即使软件不进行商业化，其开发所花费的经济成本也不高，都是在一个人可以承受的范围内；而若进行商业化并得到一定的青睐，那么软件系统也可以带来经济收入，那么相对于前期的支出，软件系统是可以获得收益的。

2.2 社会可行性

这套基于遗传算法编程实现的排课系统是个人通过 Java 自主编程实现，在当今社会的教育发展现状下，也恰巧是需要更多这么实用性的软件系统的，况且当下环境中，虽然有很多教育机构都有他们自己的定制软件，亦或者说有很多专注排课软件研发的公司，针对不同的教学生态、教学模式下研发的不同的软件，而后根据不同机构的需求将其软件授权给这些机构使用，但是这授权费用也是相对较高高，这也通常导致很多小型的机构，创业型机构对于这笔费用的支出处于犹豫状态；而我们提供的排课系统前期完全免费并开源，可以实现二次开发，实现定制化。

对于社会法律、政策等条件，我们的软件也是完全没有任何敏感的元素会触犯相关的法律法规或者是政策条文，反而在当下的教育行业蒸蒸日上的社会环境中，有着推动行业发展的作用，虽然其作用很微小。但其更大的作用还是给教育工作者带来了很大的便利，大大提高了教育机构安排课程的效率，让教育教学工作进行得更加有条不紊。

在社会推广方面，软件本身得益于其实用性以及高效性，也将不但不触犯任何法律条文，反而其可靠性、实用性也将会得到人们的青睐；软件由于其自主性也不惧怕任何侵权相关的问题。

综合以上各方面的社会因素分析，排课系统的开发在社会可行性上行得通。

2.3 技术可行性

该排课系统使用了当前软件行业中比较主流的技术架构，即 B/S 架构，也就是浏览器/服务器架构，使用这种架构极大地提升了用户体验，不再像传统那样需要使用某一个系统还需要安装一个客户端(Client)才可以，如今不管是手机还是计算机都有浏览器这个软件，只要有浏览器我们就可以使用系统。

系统使用了摘要中所述的框架进行搭建，这些框架可以快速成型项目结构，不需要像传统的项目一样写很多繁琐的配置文件，可以简化开发，提高开发效率，在计划的时

间内可以完成软件系统的编写。

系统在主流框架的整合下，变得更加容易维护，相对来说系统的稳定性是很强的，但是系统稳定运行的条件通常取决于其所部署的环境，因此我们只要选择比较稳定的服务器，并且选择的服务器能支撑多数情况下系统的并发量，通常来说系统也是稳定运行；软件质量上，在规范编码的加持下，系统的实现代码逻辑也有条有理，符合逻辑，并且在系统完成后也进行了很多的测试，因此软件质量是比较可靠的。

之前所提到的，在当下社会科技比较发达的情况下，计算机硬件的配置也相当高，价格也比较便宜，人们通常也可以支付得起这一笔费用，因而说只要软件运行了，那么也无法吃满当下常见的计算机资源，由于代码的执行效率高，因此在实际生产环境中，该系统可以完全可以应对人们的需求，给人以良好的体验，生产效率也相当优秀。

综上所述，系统开发技术可行。

3 需求分析

3.1 任务概述

3.1.1 目标

科技发展，社会进步以及“科教兴国、人才强国”战略的推动作用，使得如今的教育教学资源直线上升，在课程安排的工作上也因为课程增加，个性化条件增加带来了许多的排课困难，不再像传统那样可以通过人工手动去实现课程编排，即使可以，那么也将付出巨大的代价。

因此我们本次需要研发出来一款软件来实现排课工作，帮助教育工作者提高工作效率，不让他们在课程安排工作中花费太多本来不应该花费的时间。软件系统的最基本功能应该是能够实现排课，这也是最核心的部分。其次我们希望该系统可以实现信息管理功能，比如知道学校学生的一些基本信息以及对这些信息的修改，这些信息管理包含教室管理，讲师管理，学生管理，课程管理，以上为必须实现的基本需求；除此之外我们系统仍可以在开发过程中根据实际情况进行相应地添加一些实用的功能。

3.1.2 假定和约束

对该系统的开发而言，暂时只对其做出相应的约束即可，限定系统的开发语言为 Java 1.8；使用 MySQL 8.x 的数据库；后端框架采用当前主流的 Spring Boot 框架；前端通过 Element UI 组件与 Vue 框架整合搭建前端页面，如果有文件需要云存储则选择七牛云或者阿里云的云空间来存储；数据库持久层框架使用 Mybatis-Plus，其与 Spring Boot 框架将可以完美地结合使用，实现对数据库的操作；数据可视化使用百度官方的 Echarts

实现对系统数据的查看；Excel 文件的导入导出使用 Easypoi 实现。

3.1.3 用户特点

该系统的目标用户为培训机构中的师生，预设的用户为学生、讲师、教务处主任即管理员，这些用户对于计算机软件了解程度通常来说比较低，他们所关注的层面只是如何去使用该系统来满足他们的需求。当然也可以用于其它教学生态与系统设定的符合的教育机构，即使如此，系统的用户依然是属于教育工作者以及学生。

3.2 需求规定

3.2.1 对功能的规定

系统应该拥有起码 3 种用户，学生、讲师和管理员(教务处主任)，教务处主任拥有最高权限，讲师次之。

管理员权限：

- (1) 查看系统数据
- (2) 排课管理
- (3) 课程管理
- (4) 讲师管理
- (5) 学生管理
- (6) 班级管理
- (7) 教学资料管理
- (8) 教学设施管理
- (9) 登录、注册

讲师权限：

- (1) 查看系统数据
- (2) 查看课表
- (3) 班级管理
- (4) 学生管理
- (5) 课程管理
- (6) 教学资料管理
- (7) 登录、注册

学生权限：

- (1) 查看课表
- (2) 在线学习
- (3) 加入班级

- (4) 在线自测
- (5) 查看、下载文档
- (6) 登录、注册

系统经过权限分明限制系统用户的权限后，用户无法越过自己的权力范围去操作别的数据，这样可在很大程度上增强系统的安全性，最好还可以通过加密算法对用户的密码加密后再存入数据库中。

任何用户在未登录的情况下都应该无法进入系统，学生通过自行注册申请账号并以学号作为账户登录进入系统；管理员账号由系统工程师提前分配好，在这之后管理员可凭自己的账号登录系统；讲师账号经由管理员创建，因为讲师注册流程需要提交相关的资质材料给校方相关人员审核通过之后再创建账户给讲师，此后讲师才拥有系统的账户。

在系统管理员端可以现实系统的数据方便管理员查看，作为可视化的一部分，系统数据展示也将会提升用户体验。

在应用软件支持上，尽可能地兼容老版本的浏览器，以应付部分机构设备落后带来的尴尬局面。

3.2.2 对非功能的规定

当下 IT 行业高速发展，很多应用软件系统已经不再像过去那样界面丑陋，因为有很多开源的项目，让如今项目开发过程中可以选择的 UI 元素更多，因而系统的 UI 界面应该要美观简洁，能直观体现出每一个功能模块及其功能，这样用户体验才会更好；此外，系统的点击响应流畅不会发生卡顿，用户在使用过程中若系统出现小差错应该以人性化的提示展示给用户。

3.2.3 对性能的规定

当前使用的技术栈下，系统的稳定性更多地取决于系统所交付的机构方使用的服务器，服务器性能越好，当然是越能够体现出系统的流畅性，但该软件也基本上是每一家教育机构自己运行在自己购买的服务器下的，因此，系统的并发量也不会太高，这些机构并不需要购买很贵的服务器，只要系统能稳定运行即可，最基本的需求是系统能运行 4-6 个月稳定不出问题，这样才可以保证教育机构使用的整个学期过程中不会因为宕机而影响工作。

而若是系统部署在我方，用户对象为多家教育机构，此时性能的考量就需要评估每一家机构可能会出现访问系统时候的并发量再进行服务器配置的评估；服务器带宽也需要足够，保证用户点击系统的响应时间在 2 秒之内。当然也可以根据实际需求等常常无法满足使用需求的时候再进行增购服务器以实现性能的提升。

3.3 系统用例图

3.3.1 学生用例图

学生可以执行的操作有：注册、登录、更新资料、查看课表、查看或下载文档、在线学习、加入班级，学生的用例图如下所示：

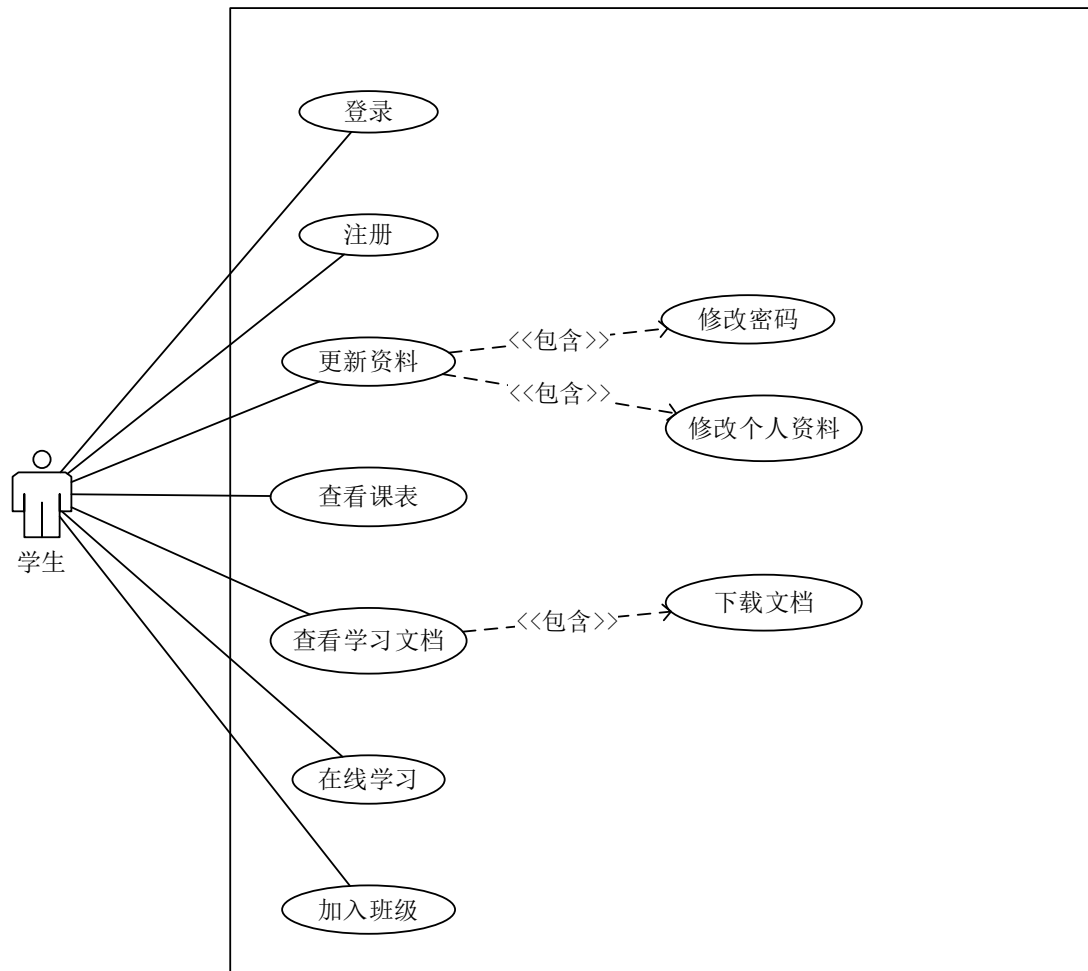


图 3.1 学生用例图

3.3.2 管理员用例图

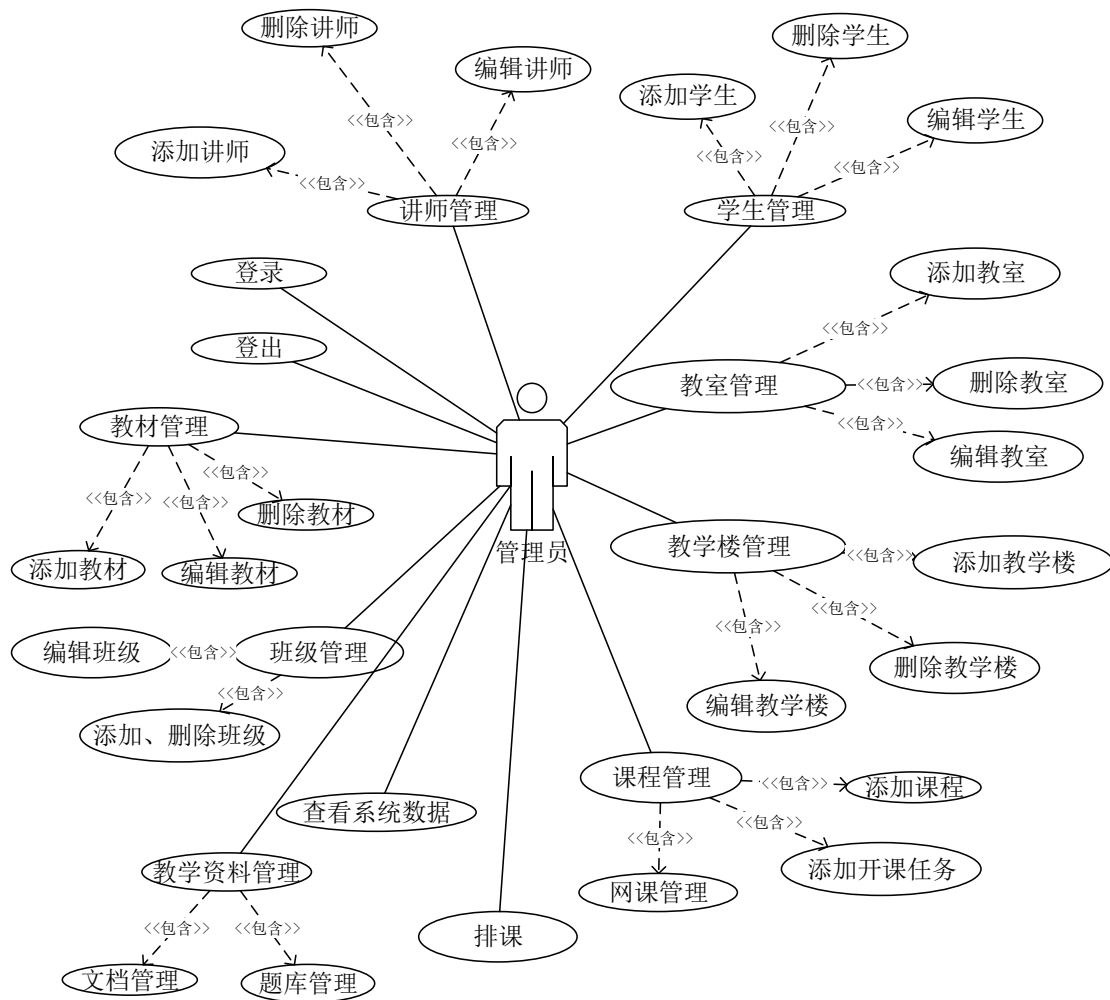


图 3.2 管理员用例图

3.3.3 讲师用例图

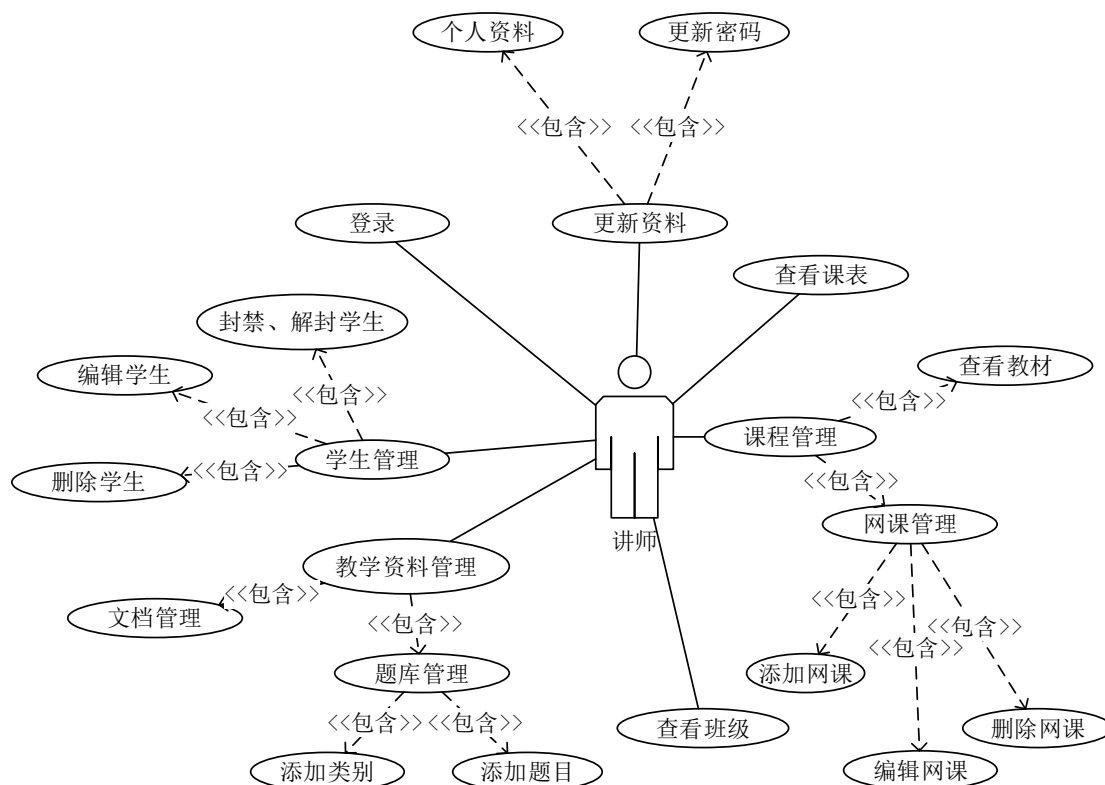


图 3.3 讲师用例图

3.4 运行设备规定

3.4.1 设备

系统需要部署在服务器上运行之后才可以给用户正常访问使用，服务器可以选择云服务器或者本地服务器；部署在 Linux 系统服务器上，也可以直接部署在人们常用的 Windows 系统的服务器上，只需要服务器上安装了相关的运行环境并做相关的部署配置即可。

3.4.2 支持的软件

如今每一台计算机或者手机都有一个上网浏览器，我们只需要浏览器即可使用该软件系统，这也是 B / S 架构开发下的便利之处，而不再需要再去单独安装一个软件使用系统。此外，为了兼容更多老版本浏览器，系统开发过程中也做出了努力，能支持到老版本的 IE6，6 以下不支持。

3.4.3 控制

系统部署后，日常维护还是需要工程师，工程师通过终端工具连接到所在服务器即可实现对项目的维护操作，数据库也部署在了项目所在的服务器，工程师也可以直接在

当前服务器对数据库进行操作；用户端只关注系统的使用。

4 总体设计

4.1 系统层次方框图

系统包含学生端，讲师端和管理员端，系统的层次方框图如下图所示：

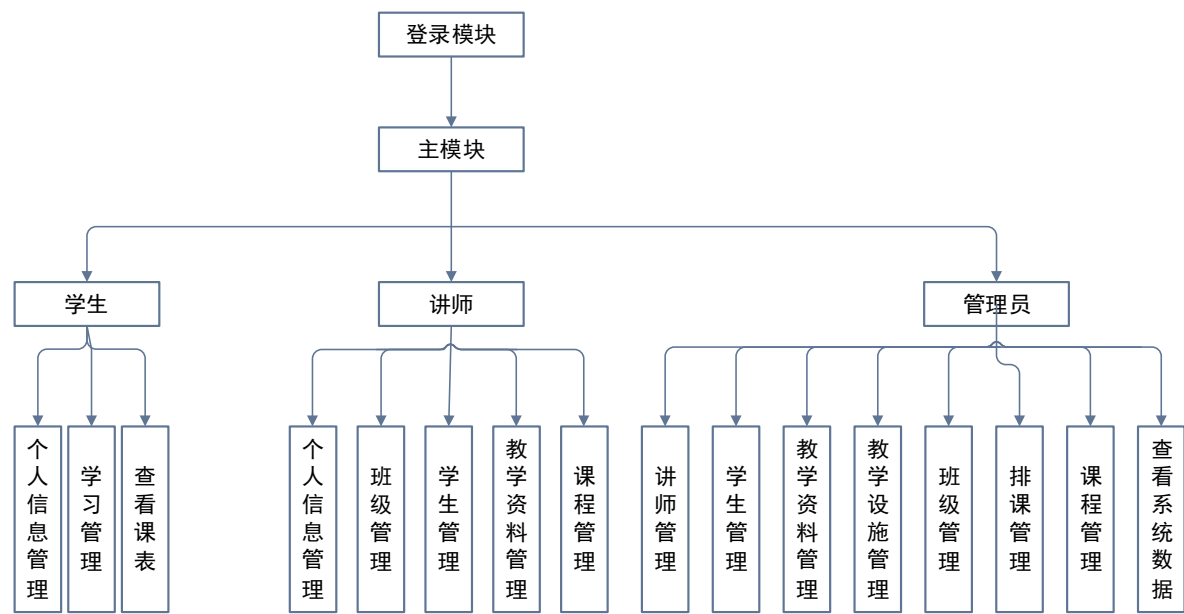


图 4.1 系统层次方框图

4.2 接口设计

4.2.1学生接口

表 4.1 学生接口表

请求路径	说明
/student/login	登录，接收学号，登录密码两个参数
/student/register	学生注册接口，接收学生注册填写的信息
/student/{id}	查询当前学生的个人信息
/student/modify	更新学生信息
/student/createn/{grade}	学生注册时给学生创建学号
/student/students/{page}	分页查询学生列表,page 为当前页
/student/search/{keyword}	根据关键字 keyword 查询学生
/student/delete/{id}	根据学生 id 删除学生
/student-class/{page}/{classno}	根据班级编号查询学生并分页显示

4.2.2 管理员接口

表 4.2 管理员接口表

请求路径	说明
/admin/login	管理员登录
/admin/{id}	根据 id 查询管理员信息
/admin/modify	更新管理员信息
/admin/password	更新管理员密码

4.2.3 讲师接口

表 4.3 讲师接口表

请求路径	说明
/teacher/login	讲师登录
/teacher/{id}	根据 id 查询讲师
/teacher/modify	更新讲师信息
/teacher/query/{page}	分页查询讲师列表
/teacher/search/{page}/{keyword}	搜索讲师并分页显示
/teacher/delete/{id}	根据 id 删除讲师
/teacher/no	创建讲师编号
/teacher/add	添加讲师
/teacher/password	更新密码
/teacher/lock/{id}	封禁、解封讲师账户

4.2.4 开课任务接口

表 4.4 开课任务接口

请求路径	说明
/classtask/{page}/{semester}	根据学期查询开课任务并分页显示
/addclasstask	手动添加开课任务
/deleteclasstask/{id}	根据 id 删除开课任务
/semester	查询所有学期
/upload	导入开课任务的 Excel 表格
/download	下载导入开课任务模板
/arrange/{semester}	将指定学期的课程任务进行排课

4.2.5教材信息接口

表 4.5 教材信息接口表

请求路径	说明
/courseinfo/{page}	分页查询教材信息
/courseinfo/add	添加教材信息
/courseinfo/delete/{id}	根据 id 删除教材
/courseinfo/modify/{id}	根据 id 更新教材
/courseinfo/search/{page}/{keyword}	关键字搜索教材

4.2.6班级接口

表 4.6 班级接口表

请求路径	说明
/class-grade/{grade}	根据年级编号查询班级列表
/addclassinfo	添加班级
/classinfo/{page}	分页查询班级信息

4.2.7教室接口

表 4.7 教室接口表

请求路径	说明
/classroom/{page}	分页查询教室列表
/classroom/delete/{id}	根据 id 删除教室
/classroom/query/{id}	根据 id 查询教室
/classroom/modify	更新教室信息
/classroom/add	添加教室

4.2.8上课计划接口

表 4.8 上课计划接口表

请求路径	说明
/courseplan/{classno}	根据班级查询课表

4.2.9文档接口

表 4.9 文档接口表

请求路径	说明
/uploaddocs	上传文档并返回在云端的 URL
/adddocs	发布新文档
/downloaddocs	下载文档
/docs/{page}	分页查看文档列表
/docs-class/{page}	查询指定班级的文档
/deletedocs	根据 id 删除文档

4.2.10练习题类型接口

表 4.10 练习题类型接口表

请求路径	说明
/exercise/categories	查询题目类别
/exercise/add	删除题目类别

4.2.11练习题接口

表 4.11 练习题接口表

请求路径	说明
/exercise/{categoryid}/{page}	根据类别查询题目并分页显示
/exercise/{page}	分页查询所有题目
/addexercise	添加题目
/exercise/train	随机出 20 道题

4.2.12教学区域接口

表 4.12 教学区域接口表

请求路径	说明
/locations/{page}	分页查询教学区域
/setteacharea	新增教学区域
/location/delete/{id}	根据 id 删除教学区域

4.2.13 教学楼接口

表 4.13 教学楼接口表

请求路径	说明
/teachbuildinfo/list/{page}	分页查询教学楼
/teachbuildinfo/list	查询所有教学楼
/teachbuildinfo/delete/{id}	根据 id 删除教学楼
/teachbuildinfo/add	添加教学楼
/teachbuildinfo/modify/{id}	更新教学楼
/teachbuildinfo/select/{id}	根据 id 查询教学楼信息

4.2.14 云存储接口

表 4.14 云存储接口表

请求路径	说明
/aliyun/video	上传网课
/aliyun/avatar	上传图片

4.2.15 系统数据接口

表 4.15 系统数据接口表

请求路径	说明
/systemdata	获取系统的数据

4.3 数据库设计

4.3.1 用户关系 E-R 图

系统的用户主要有管理员，讲师和学生，一个管理员能管理多个学生和多个讲师，一个讲师可以管理多个学生，E-R 关系图如下图所示：

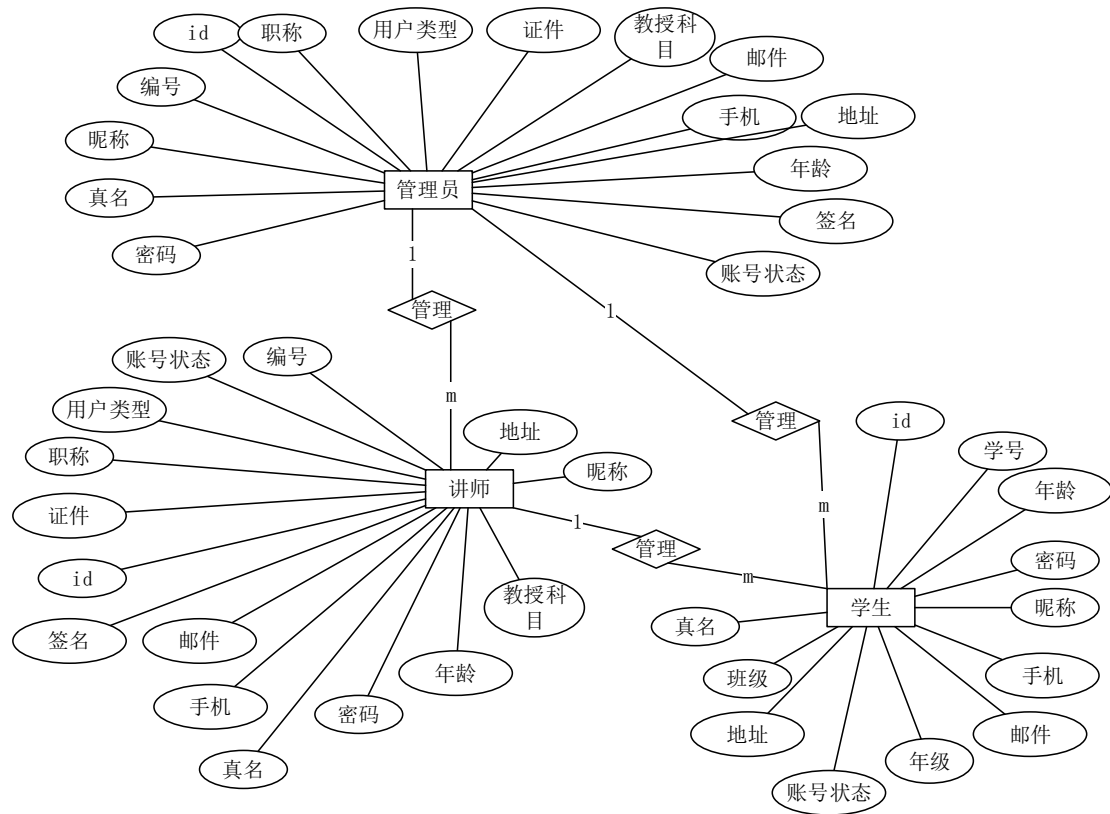


图 4.2 管理员讲师学生实体关系图

4.3.2 教学设施与年级班级学生关系 E-R 图

教学设施目前约定为教学楼，教室以及安排好的教学区域，一栋教学楼有多个教室，一个年级可以安排在多栋教学楼上上课，一个年级下面有多个班级，一个班级有多个学生，它们的关系如下图所示：

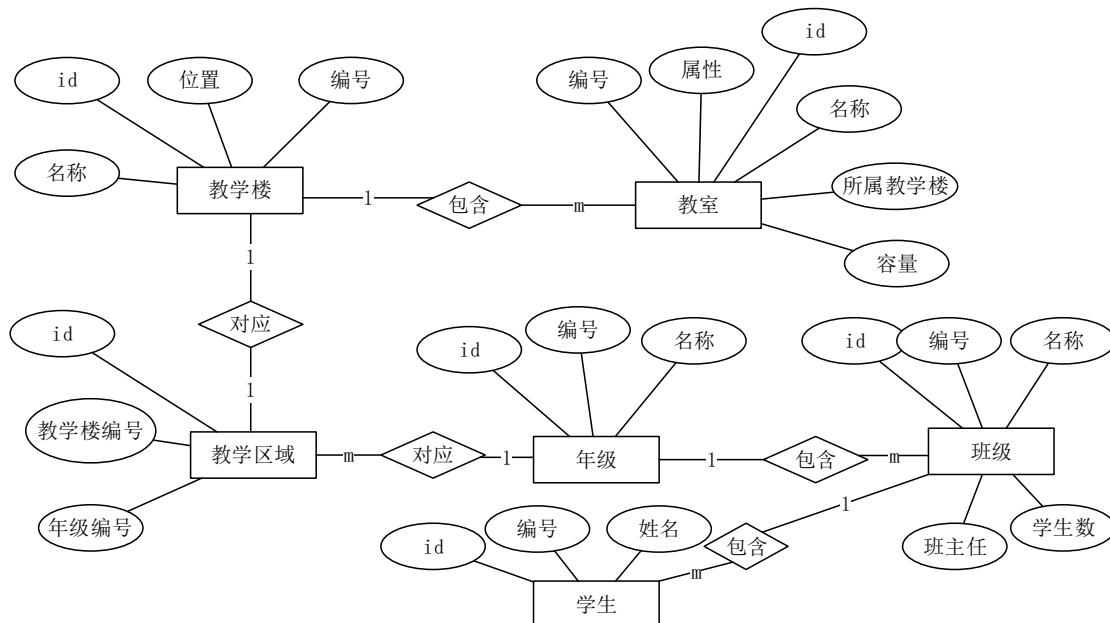


图 4.3 教学设施年级学生实体关系图

4.3.3排课任务上课任务与课程关系 E-R 图

一个排课任务对应一门课程，一门课程对应 1 到多个上课任务，上课任务是经过排课任务录入系统排课完成后才生成的，其关系如下图所示：

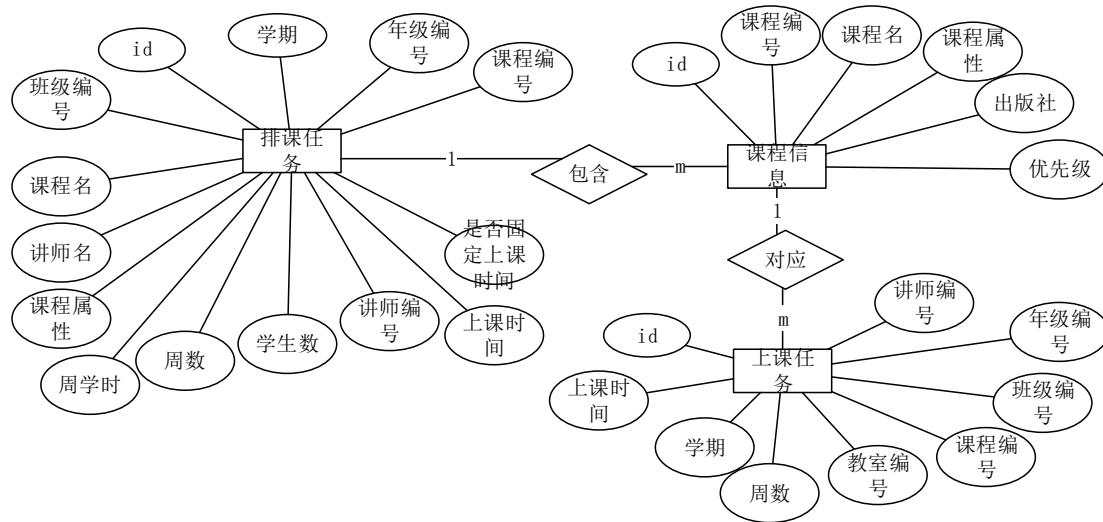


图 4.4 排课任务教材上课计划实体关系图

4.3.4习题类别与习题 E-R 模型图

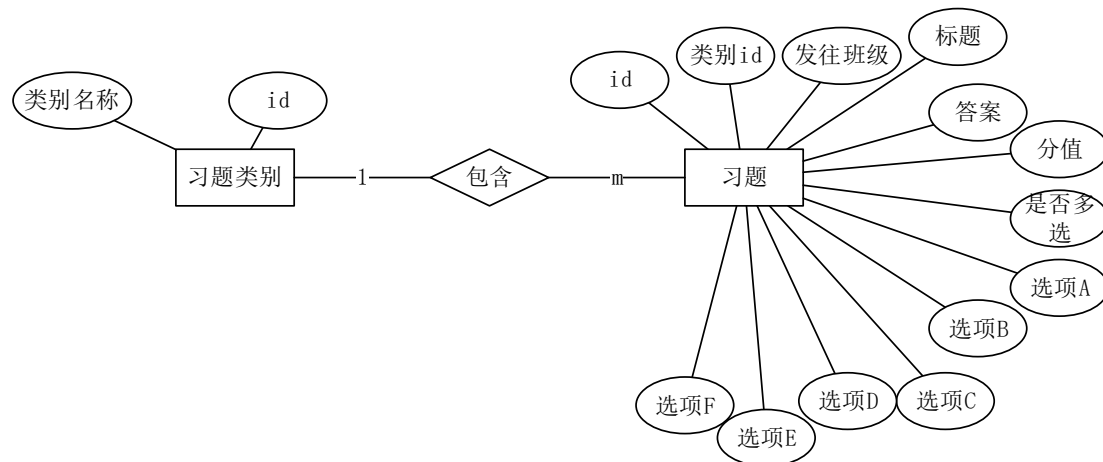


图 4.5 习题类别与习题实体关系图

4.3.5其它 E-R 图

文档 E-R 图，管理员和讲师都可以发布多个文档到指定的班级内供该班级学生查看学习，如下图所示：

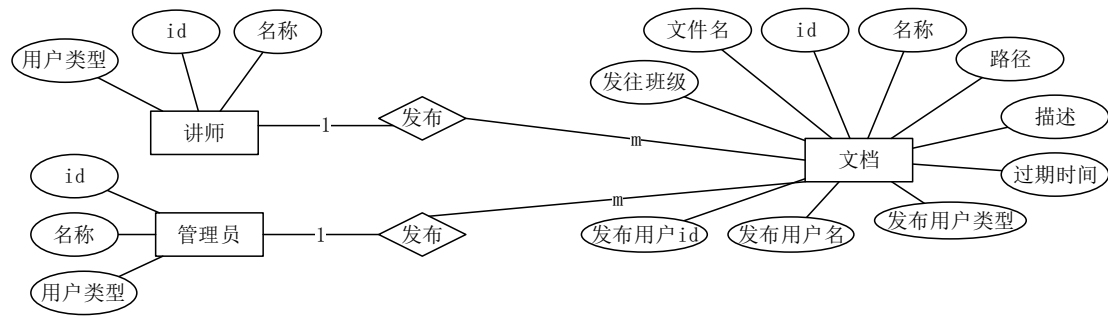


图 4.6 文档与用户关系图

学生首页轮播图如下图所示：



图 4.7 首页轮播图实体图

4.3.6 数据表设计

表 4.16 学生表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	学生 id
student_no	varchar	36		否	否	学号
username	varchar	50		否	是	昵称
password	varchar	36		否	否	密码
realname	varchar	36		否	否	真实姓名
grade	varchar	2		否	是	年级编号
class_no	varchar	36		否	是	班级编号
age	int	3		否	是	年龄
address	varchar	255		否	是	地址
telephone	varchar	50		否	是	联系电话
avatar	varchar	500		否	是	头像地址
email	varchar	50		否	是	邮件地址
description	varchar	100		否	是	个人签名
status	int	1		否	否	账号状态，0 正常，1 异常
deleted	int	1		否	否	0 不删，1 删除
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.17 讲师表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	讲师 id
teacher_no	varchar	36		否	否	讲师编号
username	varchar	50		否	是	昵称
password	varchar	36		否	否	密码
realname	varchar	36		否	否	真实姓名
jobtitle	varchar	20		否	是	职称
grade_no	varchar	2		否	是	教授年级
license	varchar	500		否	是	证件照地址
teach	varchar	10		否	是	教授科目
address	varchar	255		否	是	地址
telephone	varchar	50		否	是	联系电话
age	int	3		否	是	年龄
avatar	varchar	500		否	是	头像地址
email	varchar	50		否	是	邮件地址
power	int	1		否	是	权限
priority	int	2		否	是	优先级
description	varchar	100		否	是	个人签名
status	int	1		否	否	账号状态，默认 0 为正常，1 异常
deleted	int	1		否	否	逻辑删除默认 0 不删，1 删除
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.18 管理员(教务处主任)表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	讲师 id
admin_no	varchar	36		否	否	讲师编号
username	varchar	50		否	是	昵称
password	varchar	36		否	否	密码
realname	varchar	36		否	否	真实姓名

jobtitle	varchar	20		否	是	职称
remark	varchar	255		否	是	备注
license	varchar	500		否	是	证件照地址
teach	varchar	10		否	是	教授科目
address	varchar	255		否	是	地址
telephone	varchar	50		否	是	联系电话
age	int	3		否	是	年龄
avatar	varchar	500		否	是	头像地址
email	varchar	50		否	是	邮件地址
type	int	1		否	否	管理员类型
power	int	1		否	是	权限
priority	int	2		否	是	优先级
description	varchar	100		否	是	个人签名
status	int	1		否	否	0 正常，1 异常
deleted	int	1		否	否	默认 0 不删，1 删除
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.19 班级表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	班级 id
class_no	varchar	36		否	否	班级编号
class_name	varchar	36		否	是	班级名称
num	int	11		否	否	学生人数，默认 0
teacher	int	11		否	是	班主任，讲师 id
grade_no	varchar	2		否	否	所属年级
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.20 课程任务表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	课程任务 id
semester	varchar	20		否	是	学期
grade_no	varchar	2		否	否	年级编号

class_no	varchar	11		否	否	班级编号
course_no	varchar	36		否	否	课程编号
course_name	varchar	100		否	是	课程名
teacher_no	varchar	36		否	否	讲师编号
realname	varchar	36		否	是	讲师名称
couraeAttr	varchar	2		否	否	课程属性
studentNum	int	11		否	否	班级学生人数
weeks_sum	int	11		否	否	周数
weeks_number	int	11		否	否	周学时
isFix	int	1		否	否	是否固定上课时间， 1 不固定，2 固定
class_time	varchar	10		否	是	上课时间
deleted	int	1		否	是	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.21 教室表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	教室 id
classroom_no	varchar	36		否	否	教室编号
classroom_name	varchar	36		否	是	教室名称
teachbuild_no	varchar	4		否	否	所在教学楼编号
capacity	int	4		否	否	教室容量
attr	varchar	2		否	否	所属年级
remark	varchar	255		否	是	备注
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.22 教材信息表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	课程 id
course_no	varchar	36		否	否	课程编号
course_name	varchar	100		否	是	课程名称
course_attr	varchar	2		否	否	课程属性
publisher	varchar	255		否	是	出版社
status	int	1		否	是	状态

piority	int	2		否	是	优先级
remark	varchar	255		否	是	备注
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.23 上课计划表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	id
grade_no	varchar	2		否	否	年级编号
class_no	varchar	36		否	否	班级编号
course_no	varchar	36		否	否	课程编号
teacher_no	varchar	36		否	否	讲师编号
classroom_no	varchar	36		否	否	教室编号
class_time	varchar	10		否	否	上课时间
weeks_sum	int	5		否	是	总周数
semester	varchar	20		否	是	学期
deleted	int	1		否	是	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.24 文档信息表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	id
doc_name	varchar	255		否	否	文档名称
doc_url	varchar	1000		否	否	文档 url 地址
desc	varchar	255		否	是	文档描述
to_class_no	varchar	36		否	是	发往的班级
from_user_id	int	11	外键	否	否	发送者 id
from_user_type	int	1		否	否	发送者类型，1 管理员，2 讲师
from_user_name	varchar	36	否	否	否	发送者名称
remark	varchar	255	否	否	是	备注信息
clicks	int	11	否	否	是	点击次数

expire	int	11	否	否	是	有效天数
deleted	int	1	否	否	是	逻辑删除，默认 0
create_time	datetime	0	否	否	是	创建时间
update_time	datetime	0	否	否	是	更新时间

表 4.25 习题类型表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	类别 id
category_name	varchar	100		否	否	类别名称
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.26 习题表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	习题 id
category_id	int	11		否	否	习题类别 id
class_no	varchar	36		否	是	推送到的班级
exercise_title	varchar	255		否	否	习题题目
multiselect	int	1		否	否	默认 0 单选，1 多选
answer	varchar	12		否	否	习题答案
option_a	varchar	255		否	是	选项 A
option_b	varchar	255		否	是	选项 B
option_c	varchar	255		否	是	选项 C
option_d	varchar	255		否	是	选项 D
option_e	varchar	255		否	是	选项 E
option_f	varchar	255		否	是	选项 F
fraction	int	3		否	是	习题分值
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.27 年级信息表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	年级 id
grade_no	varchar	2		否	否	年级编号
grade_name	varchar	36		否	否	年级名称

表 4.28 位置信息表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	id
teachbuild_no	varchar	2		否	否	教学楼编号
grade_no	varchar	2		否	否	年级编号
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

表 4.29 教学楼信息表

字段	类型	长度	约束	自增	可空	备注
id	int	11	主键	是	否	id
teach_build_no	varchar	4		否	否	教学楼编号
teach_build_name	varchar	36		否	否	教学楼名字
teache_build_location	varchar	36		否	否	教学楼位置
deleted	int	1		否	否	逻辑删除，默认 0
create_time	datetime	0		否	是	创建时间
update_time	datetime	0		否	是	更新时间

5 详细设计

5.1 注册模块

5.1.1 学生注册

学生还未拥有该系统的账号时，可以通过点击我要注册跳转到注册界面，填写好自己的基本个人信息，通过选择自己当前的年级后，通过点击右侧的生成学号按钮向后台

发起请求给学生生成一个对应年级的随机学号，后台返回给前台的学号会以禁止编辑的形式展示在前端，必填项录入完毕后点击注册按钮提交表单到后台完成注册，注册成功之后通过学号，密码登录系统。

5.1.2 讲师注册

目前系统设定为讲师将相关的资质材料交给校方审核后，由校方工作人员，例如教务处主任将帮讲师添加账号，讲师编号也是系统自动生成，账号添加完毕时默认设置密码为 123456，讲师通过讲师编号和密码在讲师入口处登录成功后可以修改完善讲师的个人信息。

用户注册的时序图如下所示：

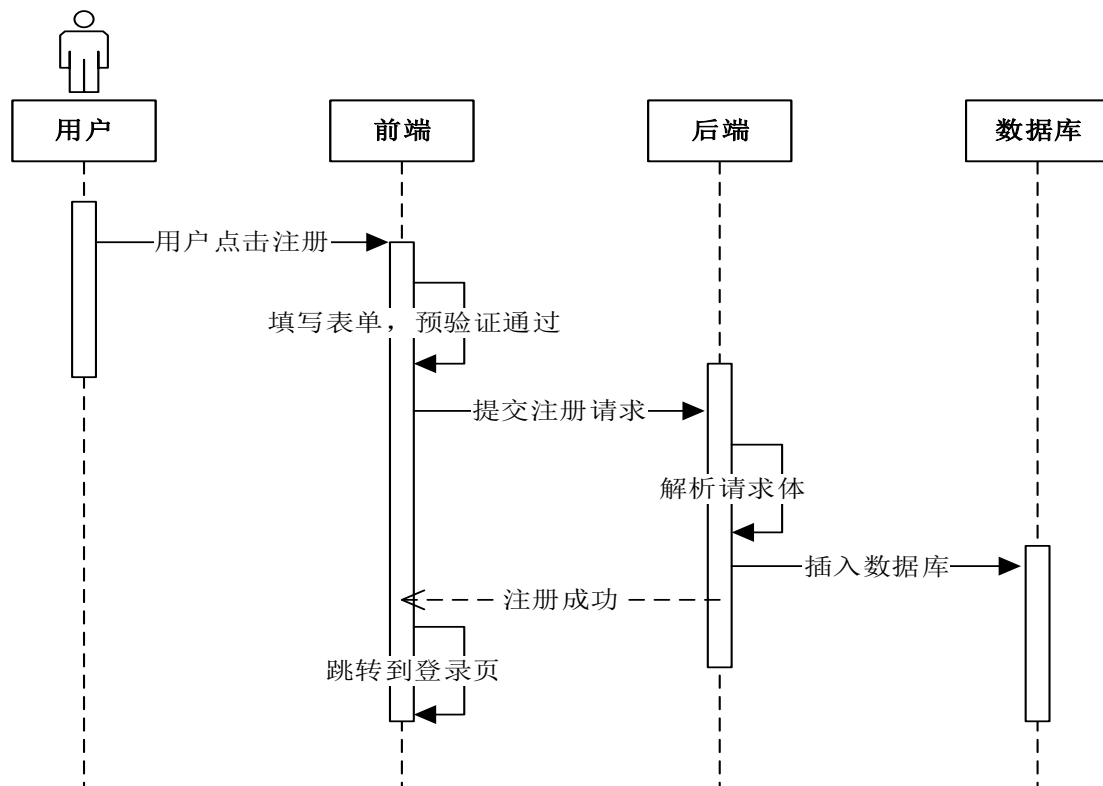


图 5.1 学生注册时序图

5.2 登录模块

5.2.1 用户登录

用户(学生，讲师，管理员)在登录页面输入自己的账号以及密码，点击登录将表单数据提交到后台系统，系统会将学生提交的账号密码与数据库的学生数据匹配，匹配通过则登录成功，进入系统并能执行相应的操作；否则继续停留在登录界面并给出登录失败的提示。用户登录时序图如下所示：

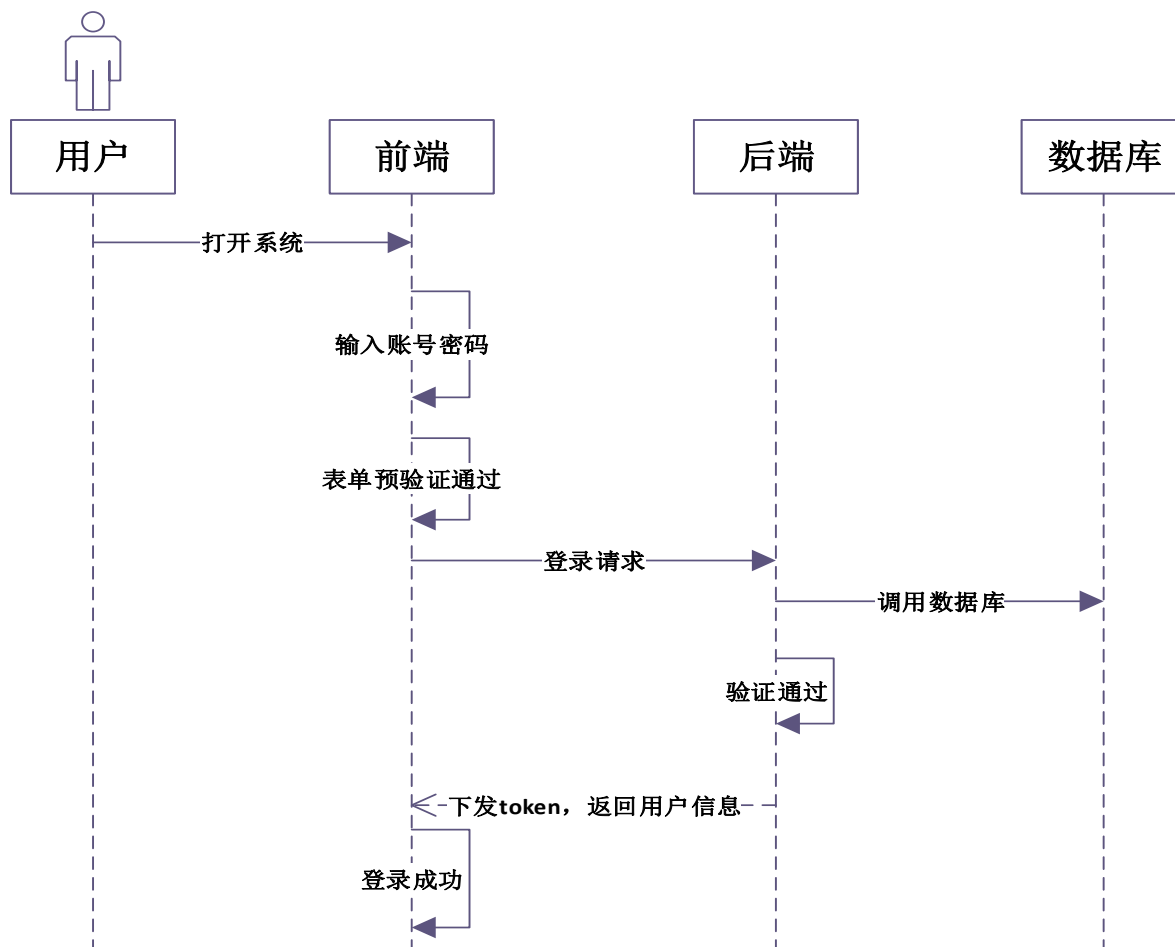


图 5.2 用户登录时序图

5.3 修改用户信息

5.3.1 修改信息

系统的任何一种用户登录成功后，其个人信息将会存储在浏览器端，当用户需要修改个人信息时，首先从浏览器端存储的信息中获取用户的本地信息并在前端显示，此时用户可以在表单中重新编辑自己的个人信息，修改完成后再提交新的表单给后端修改数据库信息，完成后浏览器端返回用户的新信息，并在浏览器端更新 `localStorage` 中新的用户信息。用户修改信息时序图如下图所示：

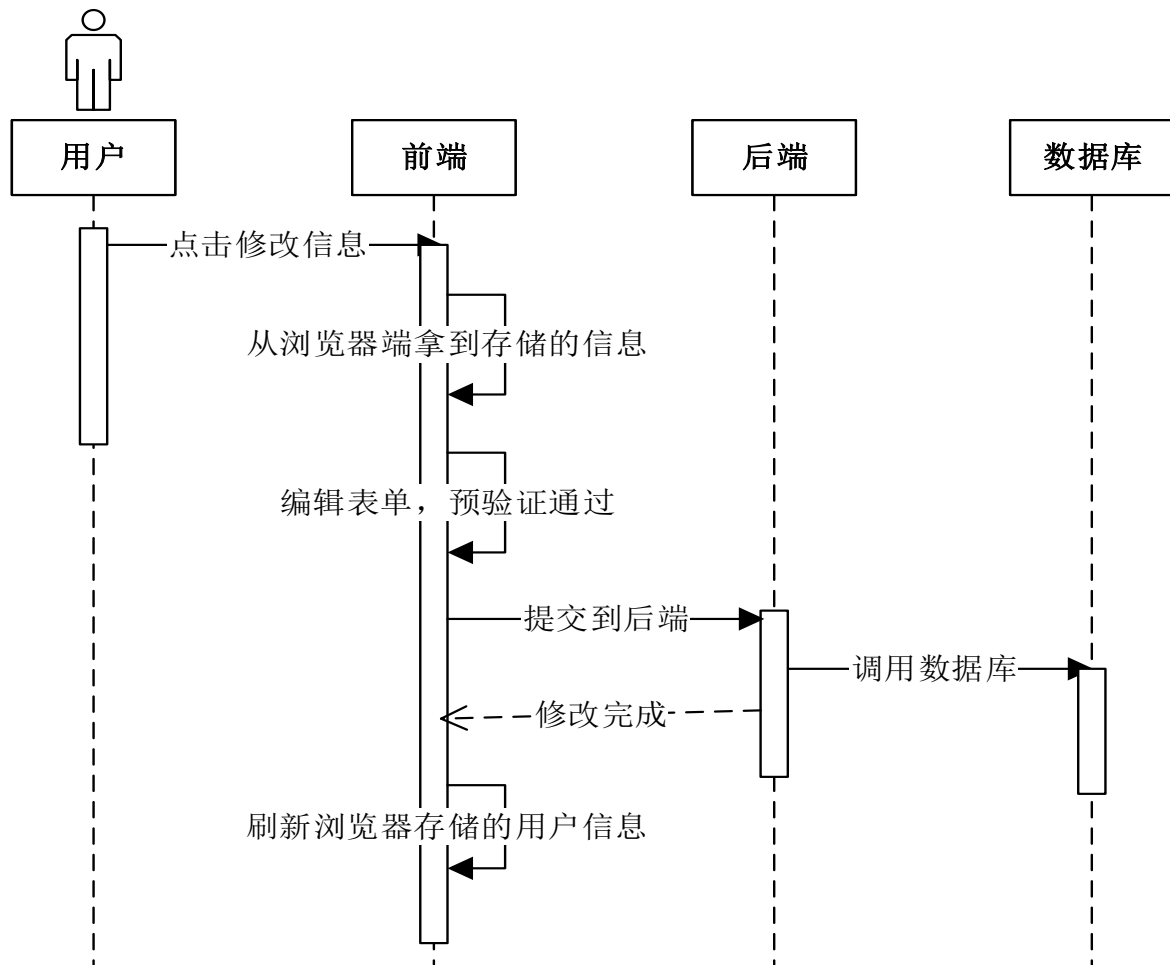


图 5.3 修改用户信息时序图

5.3.2 修改密码

用户点击修改密码时，需要填写旧密码以及输入两次新密码进行确认后，前端预验证通过后(即两次新密码一致以及密码长度达到要求)，点击提交按钮发送请求到后端，首先需要验证用户的旧密码是否正确，如果正确即可进行修改用户的密码，并让用户下线重新登录；否则返回旧密码错误的提示信息给前端。用户修改密码的时序图如下所示：

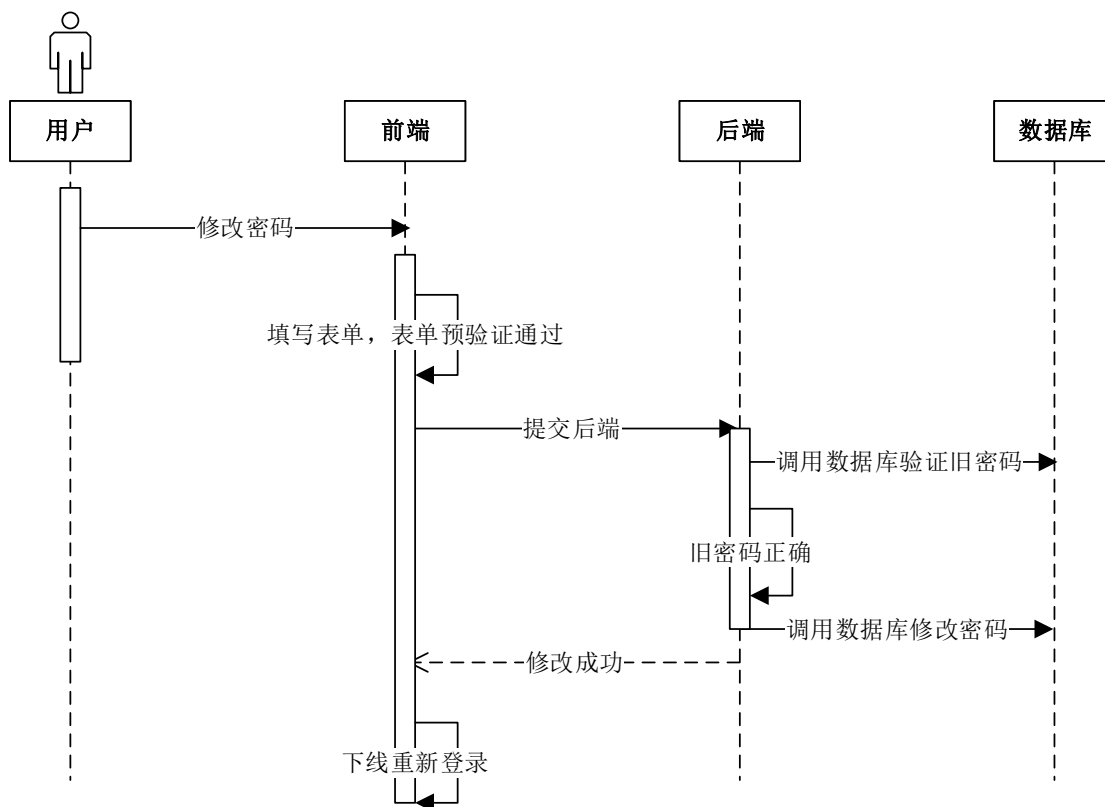


图 5.4 修改密码时序图

5.4 课程管理

5.4.1 网课管理

学生端的首页就是网站的前端学习页面，在首页学生可以查看学校开设的网络课程，学生在课后时间可以通过网上学习网课的方式进行学习，点击相应的网课之后会跳转到网课的详细信息页面，点击相应的视频即可播放该视频来学习。

5.4.2 教材管理

学校日常开展的教学工作所使用的教材都可以放在教材管理处进行统一管理，在增设排课任务时从教材列表中选取相应的教材编号作为课程编号添加到排课任务中。教材管理处可以实现教材的搜索，编辑，删除以及添加。下图是添加、编辑教材信息的时序图：

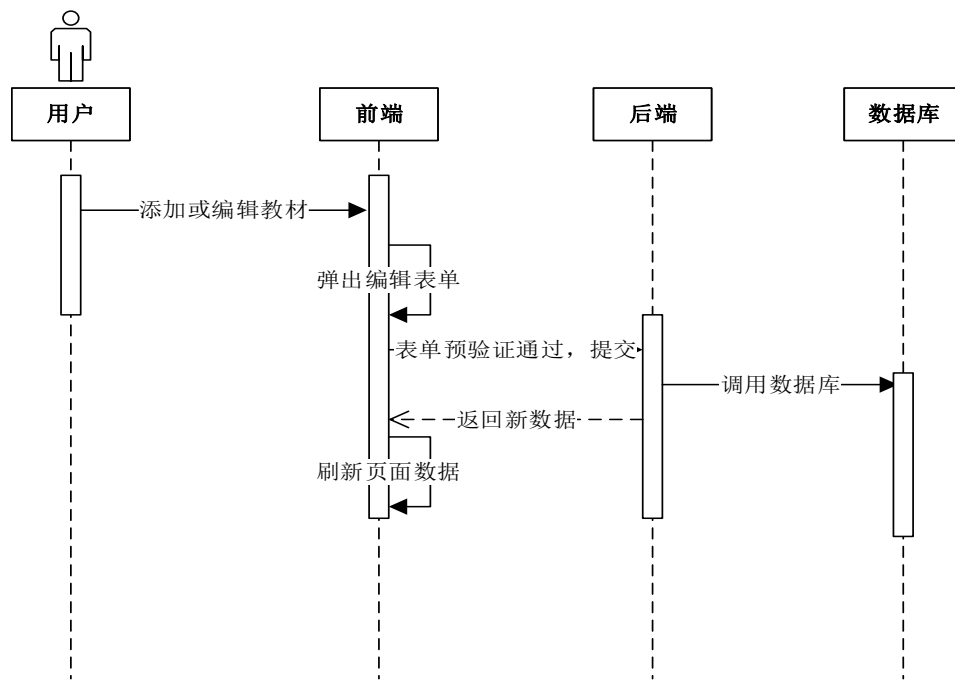


图 5.5 添加教材时序图

5.5 排课模块

5.5.1 下载排课任务 Excel 模板

将 Excel 模板存放在某一路径下，当用管理员点击下载模板时，调用下载模板接口，后端将文件返回给前端，并自动下载到用户的计算机上。

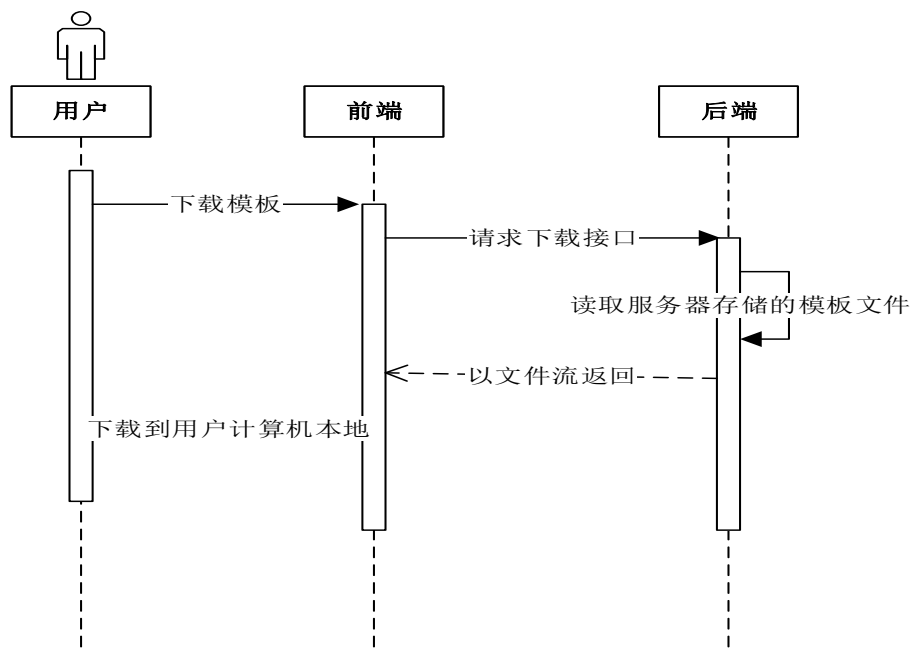


图 5.6 下载 Excel 模板时序图

5.5.2 Excel 导入排课任务

用户将相应的内容按照要求填写到 Excel 模板中之后点击选择文件，选择目标.xlsx 或者.xls 文件，点击上传按钮，后端解析文件中的内容并存储到数据库中即可完成排课任务的添加工作。

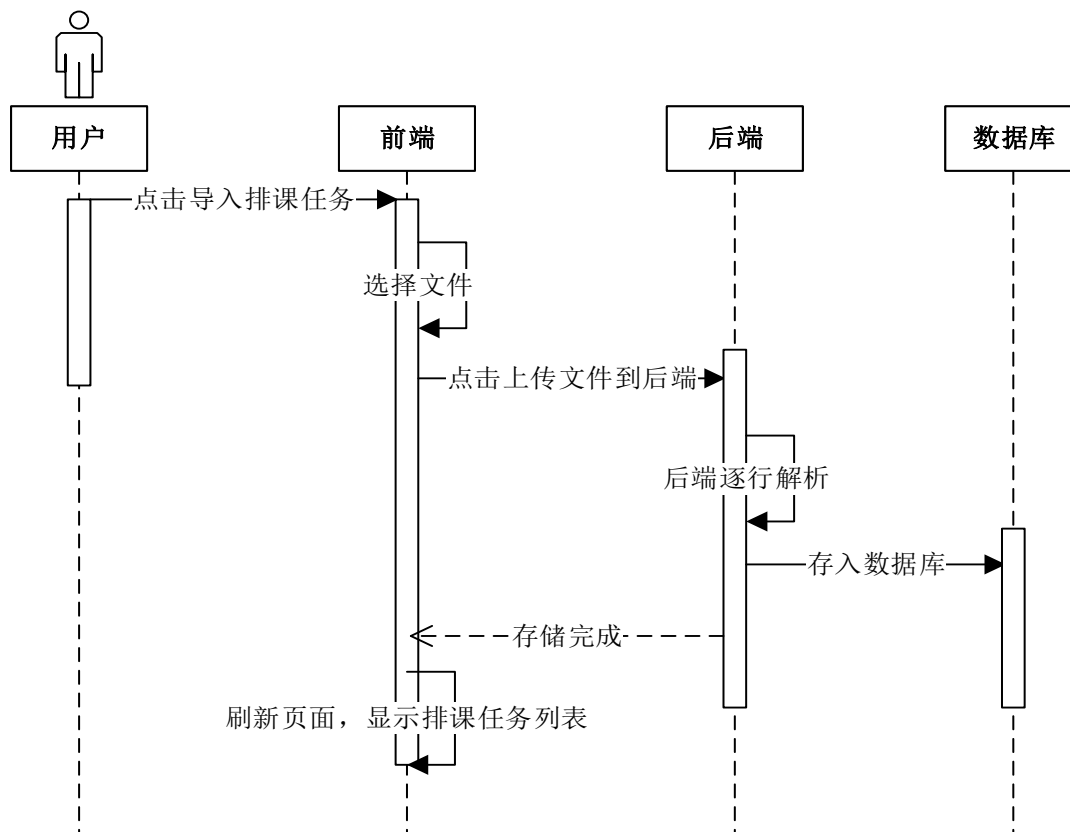


图 5.7 Excel 导入排课任务时序图

5.5.3 手动添加排课任务

如果针对少数或者追加课程任务的情况，用户可以点击手动添加，则弹出表单，用户手工填写好表单需要的信息，待前端预验证通过之后，点击提交按钮即可将相关的信息传送到后端并经程序处理存入数据库，一样可以添加排课任务。

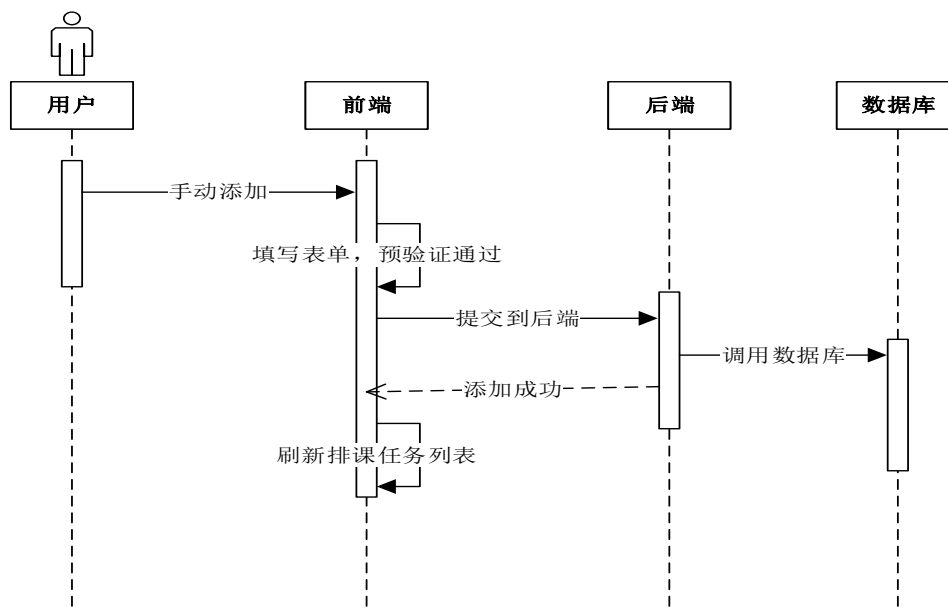


图 5.8 手动添加排课任务时序图

5.5.4排课

在导入 Excel 数据经过后台解析之后即可实现排课任务的添加，此时可以在前端看到所有的任务，点击上方按钮之后前端将选中的学期传送到后端，后台接收到排课请求后，将前端携带的参数取到并用其查询排课任务表中对应本学期的任务，将任务封装成一个集合；如果获取不到任务列表，那么直接向前端返回提示信息“查询不到排课任务”；否则开始执行排课算法，排课完成之后返回成功提示并跳转到查看课表的页面；失败则给出排课失败的提示。

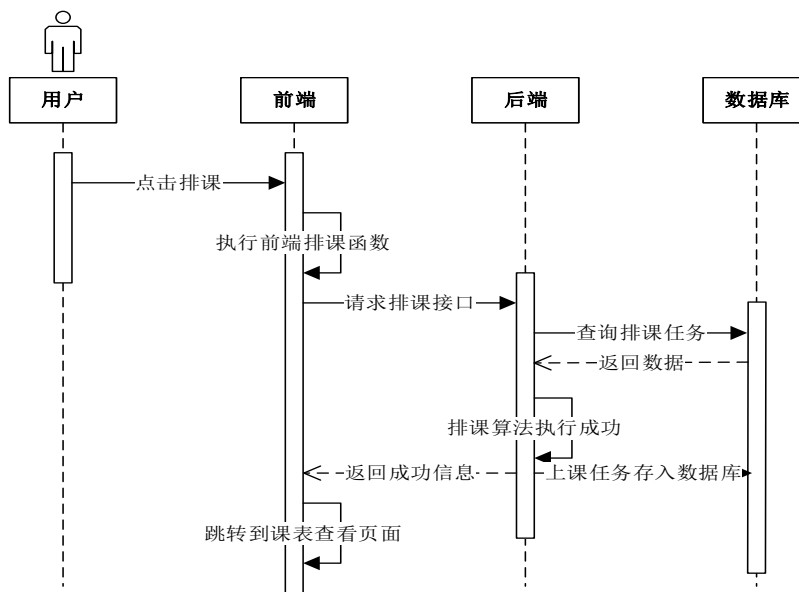


图 5.9 排课时序图

5.5.5 查看课表

每一个学生在参与教育活动前都应该根据要求加入一个属于自己的班级，可以在学生个人中心加入班级，隶属于一个班级后可以通过查看自己班级的课程表来得知自己的上课安排。

讲师和管理员可以查看所有班级的课表，对于讲师不限制其查看课表的权限，用户通过筛选年级后，会向系统后台发出一个请求获得该年级下的所有班级，再选择需要查看的班级即可请求后台的数据填充在下方课程表的方格上，即完成课程表的查看。查看课表的时序图如下图所示：

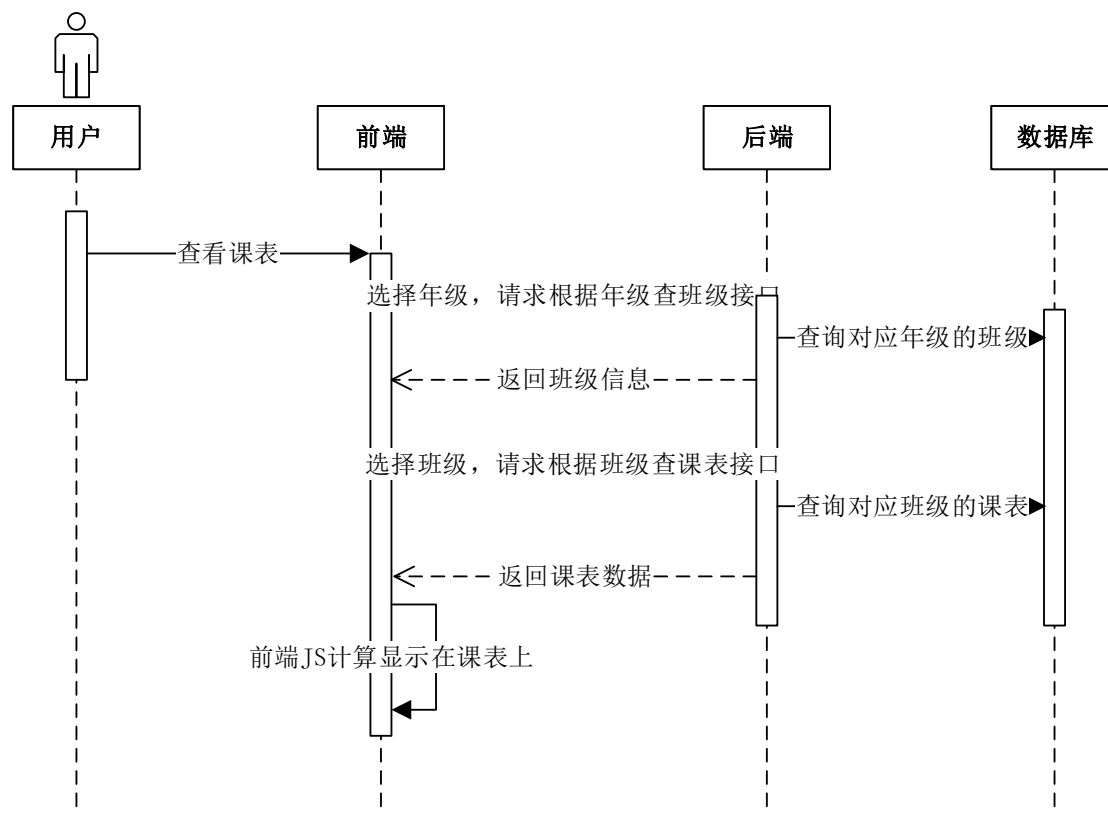


图 5.10 查看课表时序图

5.6 学生管理

在管理员和讲师端都有管理学生这个菜单，点击学生管理后即可通过列表的方式显示所有的学生信息，在这个菜单里可以通过模糊查询学生的姓名得到学生信息；还可以通过下拉选择不同年级下的不同班级获得该班级下面的学生，每一行记录后面有“编辑”和“删除”按钮，点击按钮可以完成相应的操作。编辑学生时序图如下图所示：

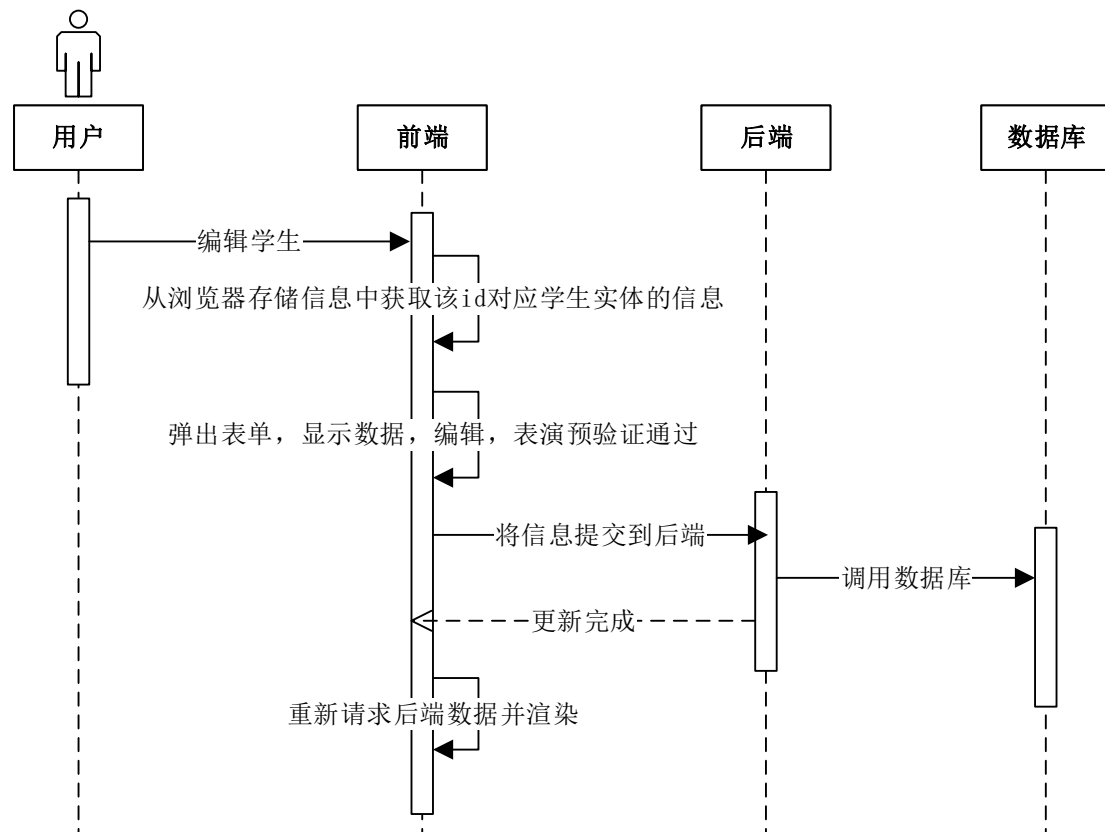


图 5.11 编辑学生时序图

5.7 讲师管理

在管理员端可以对讲师进行管理，包括添加讲师，删除讲师以及编辑讲师的个人信息。添加讲师在讲师注册处已经说明，编辑讲师的时序与编辑学生的一样，都是从本地 `localStorage` 中获取对应行的实体信息直接进行修改后提交到后端根据 `id` 更新再返回前端，这里不再贴图。

5.8 教学资料管理

讲师可以将学习文档或者作业以 `doc`，`docx`，`pdf` 等文件的格式上传到系统，选择指定的班级即可将该文件推送到目标的班级，随后该班级的学生即可在登录之后查看讲师发布的文档信息。在讲师端也可以查看到详细的文档列表，下载列表中的文档或者在线预览文档。

5.8.1 在线测试题库管理

当前系统支持的在线测试类型只有选择题，题目可以分类在不同的类别下面，讲师也可以在当前页面通过下拉选择不同的题目类型来查看该类型下面的题目；讲师可以添加题库的类型，在添加新的题库到题库时需要选择该题目所属的题目类别后点击提交才

可以添加新题库到题库中。如果需要新增题目类型可以点击上方的新增类型按钮添加新的题目类型，方便进行题库的管理；新增题目的时序图如下图所示：

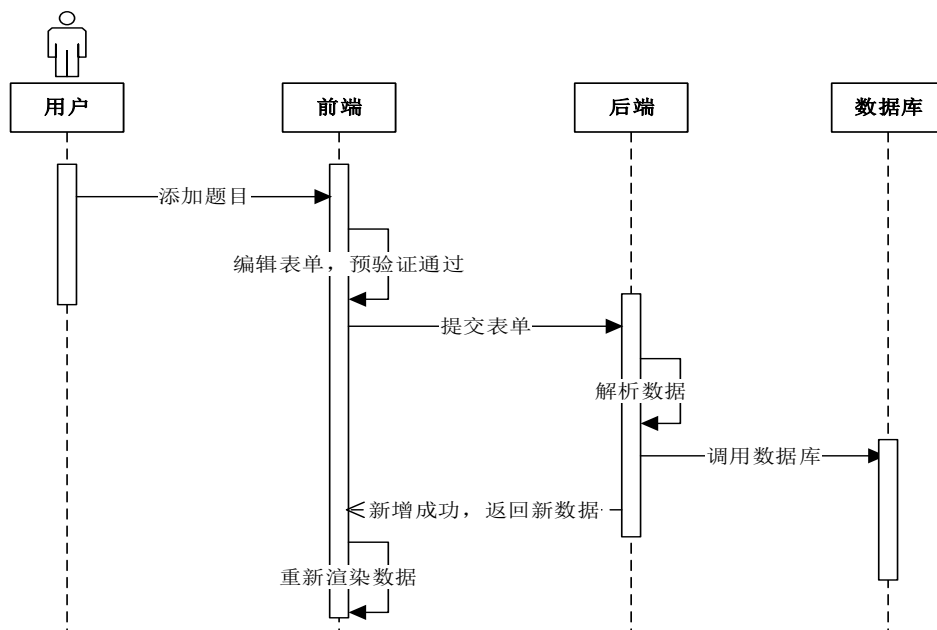


图 5.12 新增习题时序图

5.8.2 作业学习文档管理

讲师或者管理员可以在后台上传相关的文档或者其它文件供学生端下载查看，发布学生的作业也可以通过上传文档的形式上传到指定的班级供指定班级的学生查看。上传文档时序图如下图所示：

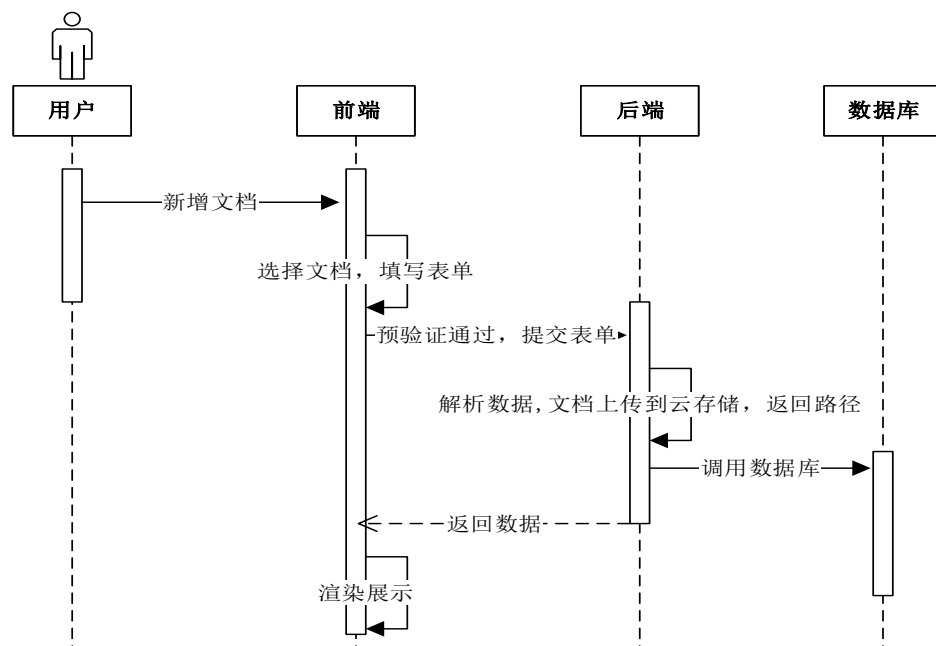


图 5.13 发布文档时序图

5.9 班级管理

5.9.1 查看班级

管理员在班级管理页面除了可以像讲师那样查看当前学校不同年级以下的不同班级以外还可以给不同的年级新增班级，新增班级的时候可以选择学校的讲师中的一个作为该班级的班主任。

5.9.2 添加班级

在班级列表页面点击新增班级之后弹出表单，选择年级，选择班主任之后开始填写班级的相应信息，填写完毕后点击提交到系统后台经过存储数据即可完成班级的添加。

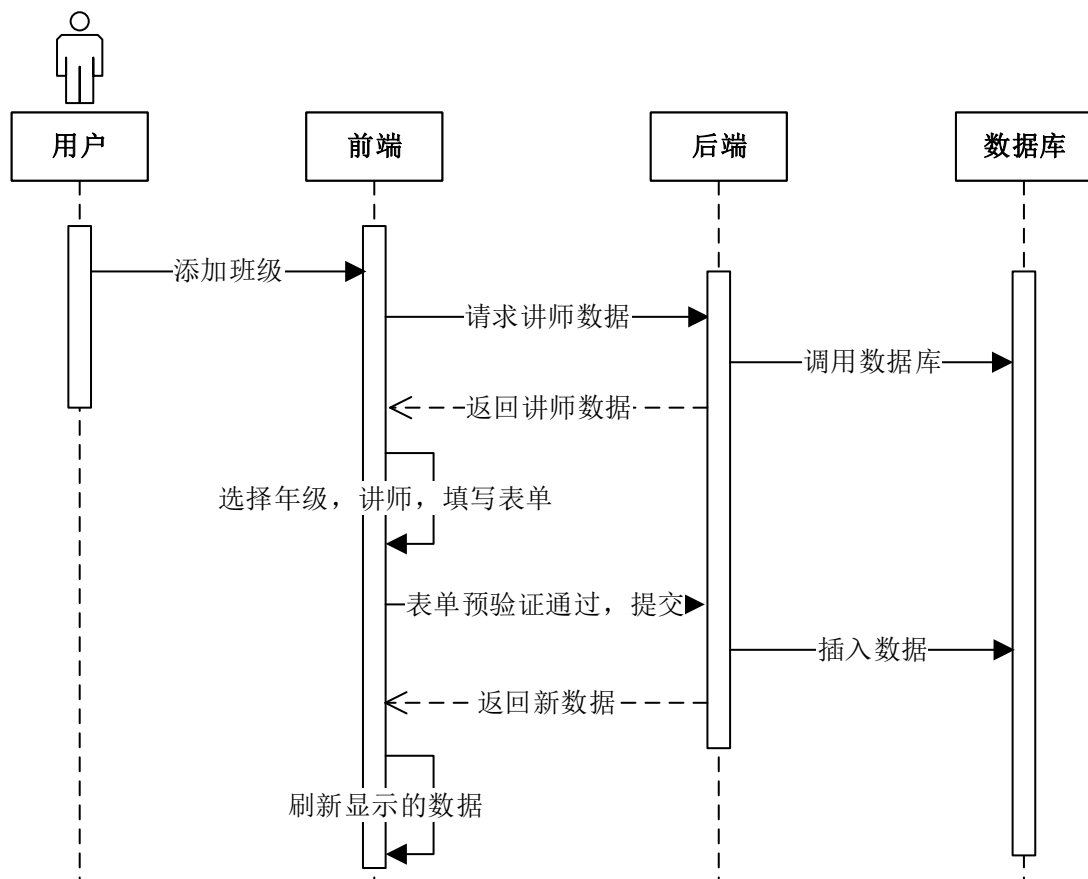


图 5.14 添加班级时序图

5.10 教学设施管理

教学设施主要就是平时举办教学活动中所使用到的教学楼，教室等。

5.10.1教学楼管理

教学楼信息包含教学楼编号，名称，所在区域，管理员可以删除，添加，编辑教学楼，添加教学楼时序图如下所示：

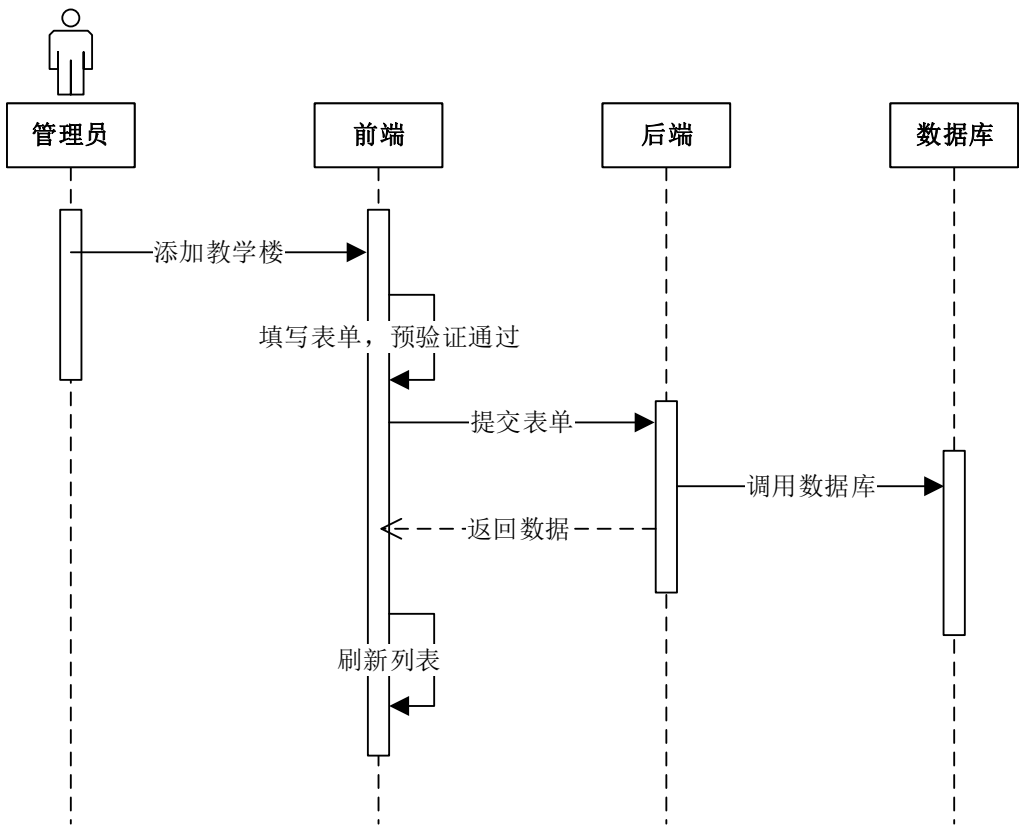


图 5.15 添加教学楼时序图

5.10.2教室管理

管理员点击教室管理菜单后，右侧界面将以表格的形式列出所有教学楼下的教室，可以添加，编辑和删除教室。添加教室的时序图如下所示：

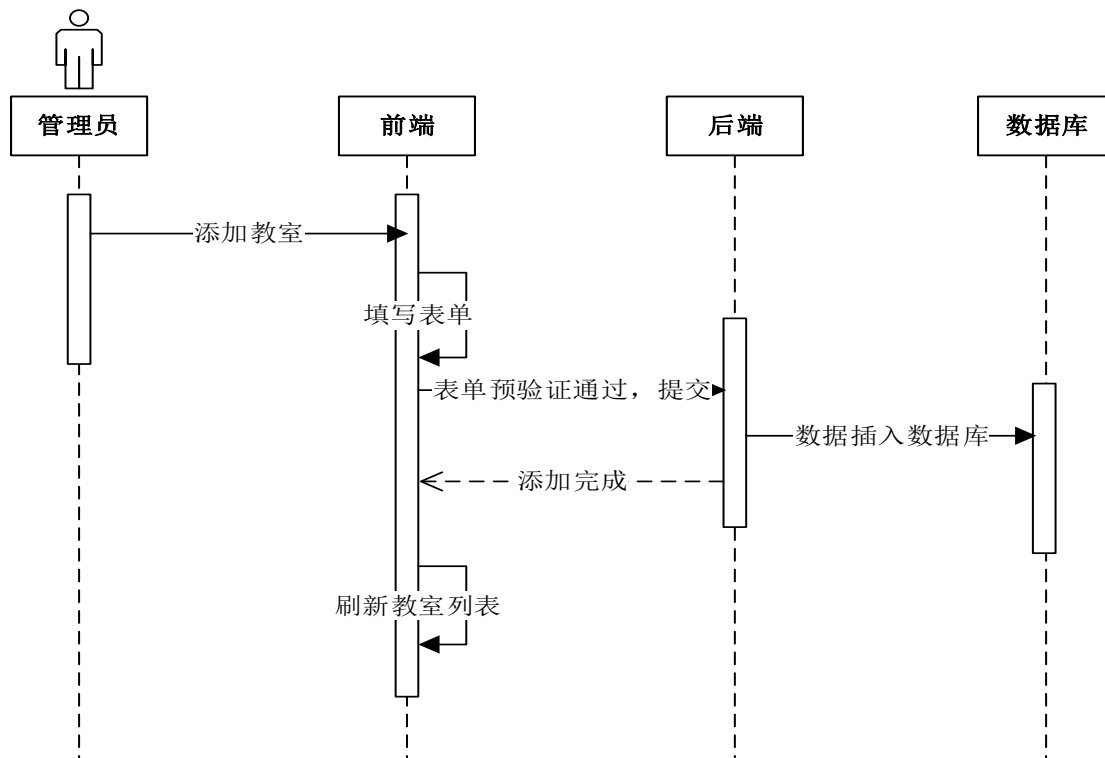


图 5.16 添加教室时序图

5.10.3 教学区域管理

教学区域是用于设定某一个在哪些教学楼栋中上课，设定好之后，排课的时候分配教室就可以通过教学区域表去查找教室分配给对应年级下面的班级，添加教学区域时候通过下拉选择年级以及下拉选择教学楼来提交到后台实现添加，时序图与以上添加操作的时序基本一致。

5.11 学生端功能

学生进入系统后进入系统首页，即网课页面，在此页面学生可以通过登录之后查看播放相应的网课视频；学生注册账号后默认没有班级，需要学生自行加入班级才可以，有了班级之后可以查看班级发布的作业或者其它一些学习文档；此外学生还可以自己在线自测，系统会随机出 20 道选择题给学生答题。

6 系统实现

6.1 注册模块

6.1.1 学生注册

学生可以自行注册账号，学生在注册页面需要填写昵称、密码、真实姓名、选择自己所属的年级之后点击学号后面的创建学号会向后端发送请求根据年份以及学生选择的年级随机生成一个 10 位长度的学号给学生，学生可以继续选择是否完善下面的联系电话、电子邮件以及家庭地址等信息，完成信息填写之后即可点击注册按钮完成学生的注册，之后学生可以通过学号和密码登录系统。下图为学生注册的流程图：

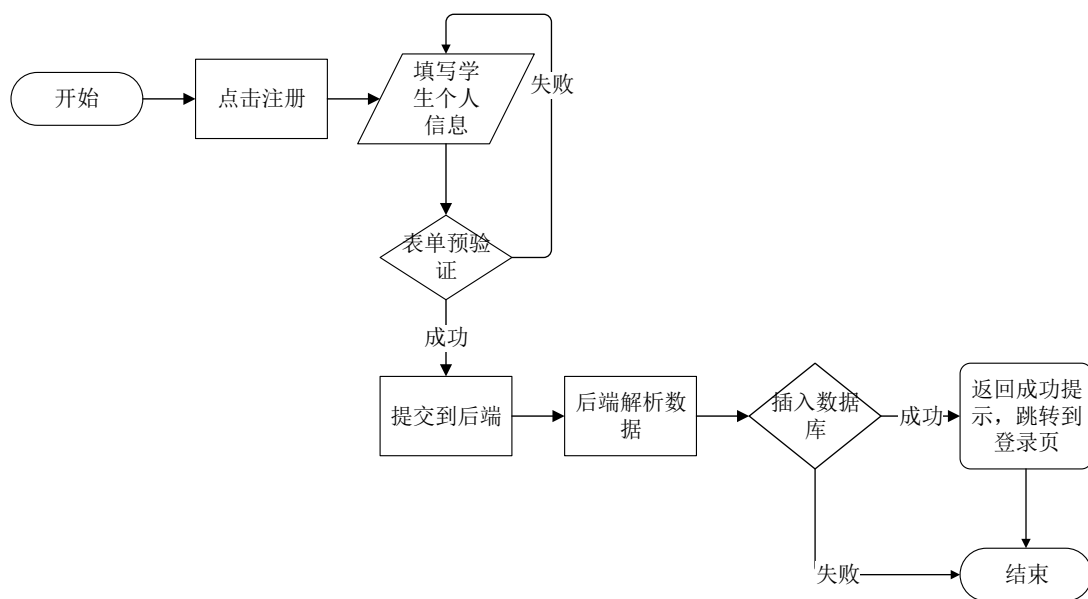


图 6.1 学生注册流程图

6.1.2 讲师注册

由于讲师需要对讲师的证件进行审核，因此需要将走正常的讲师招聘流程之后，讲师将相关的资质证件提交到校方经过校方审核通过，由校方人员将讲师的信息添加到系统中，给讲师创建一个账户之后讲师才可以使用排课系统。讲师注册流程如下图所示：

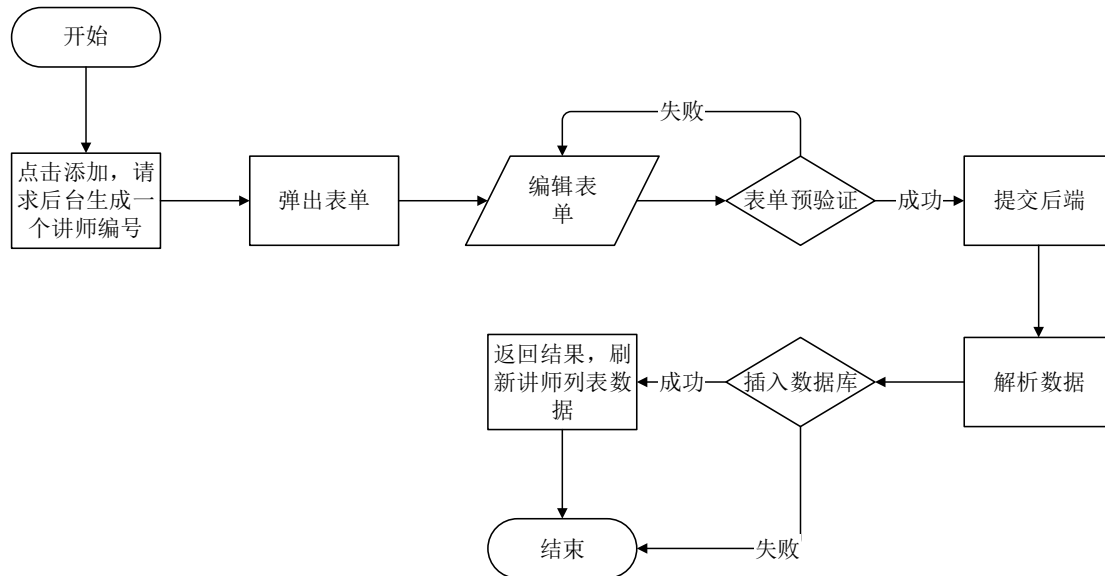


图 6.2 讲师注册流程图

6.2 登录模块

所有用户需要使用系统的功能都需要进行登录，凭借账号和密码登入系统才可以操作。用户在登录界面输入账号密码点击登录之后前端向后端发送登录请求并将用户信息携带到后端，后端获取账号与密码并于数据库匹配，匹配成功则调用 JWT 组件下发 Token 信息并返回用户信息和 token 给前端，前端拿到这些信息之后将其存储在浏览器中，登录成功，跳转到相应的界面；否则，验证账号信息不通过，后端返回错误的提示，不允许登入系统。用户登录的流程图如下图所示：

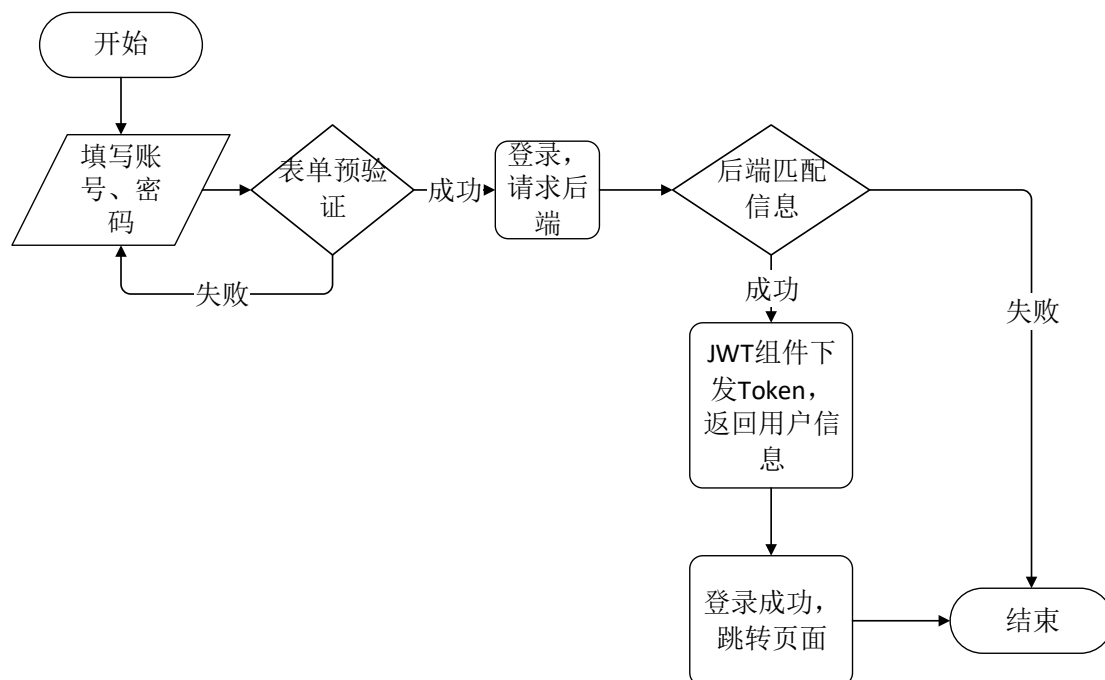


图 6.3 用户登录流程图

6.3 个人信息修改模块

由于每一个用户在登录成功后都会在自己的浏览器端存储有该用户的信息，所以在点击修改用户的信息时，只要暂时从浏览器中获取信息填充在编辑的表单上，用户编辑好新的信息之后再点击提交按钮，将新的数据提交到后端，后端即可通过信息获得需要更新信息的用户 id，直接根据 id 更新用户的信息即可；如果是修改密码那么在前端需要输入用户的旧密码以及两次新密码，提交到后台，后台根据用户的 id 去匹配其旧密码是否正确，正确之后才执行修改密码流程，否则返回错误提示到前端；用户信息修改完毕都要返回前端并刷新本地存储的信息，如果是修改密码需要下线重新登录账户。以下为修改用户信息的流程图：

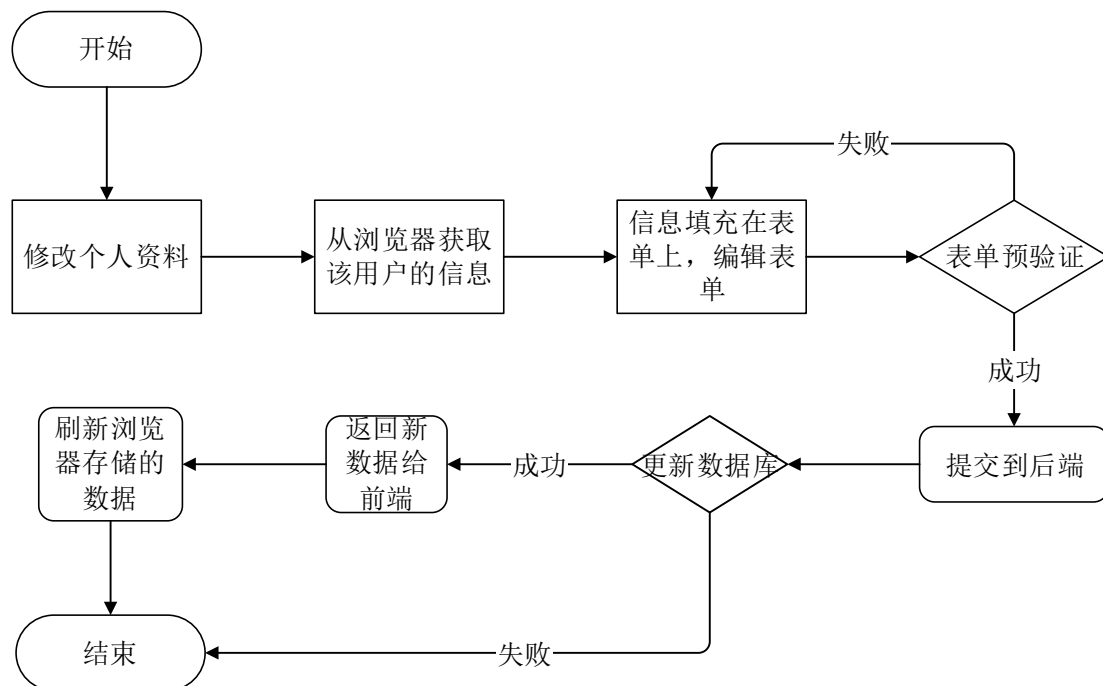


图 6.4 更新用户资料流程图

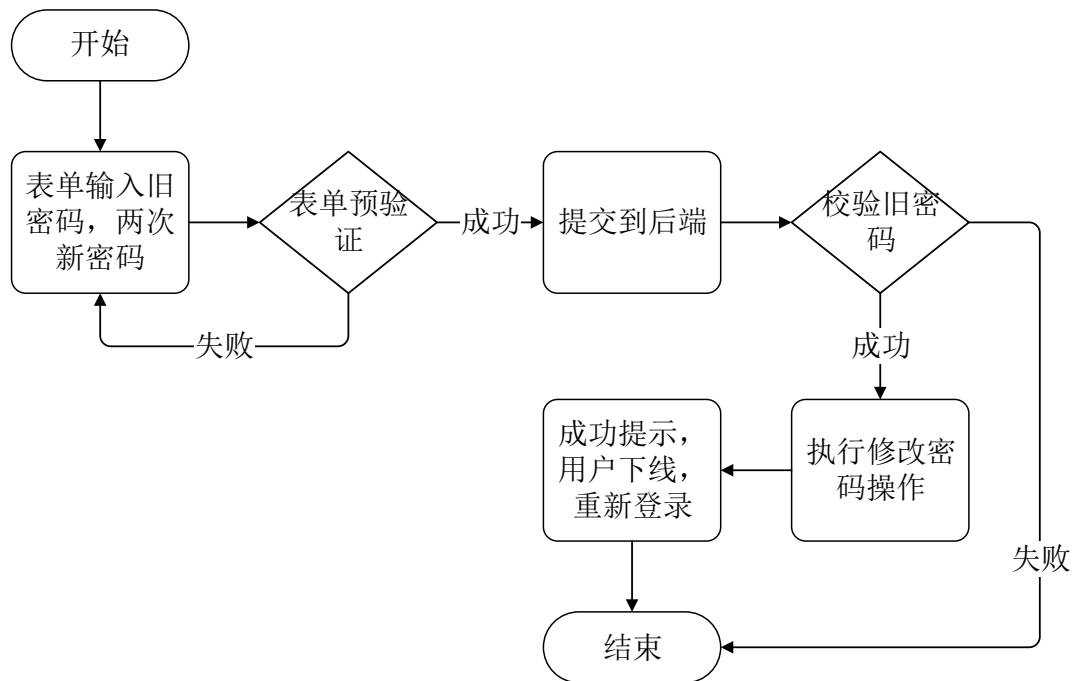


图 6.5 更新密码流程图

6.4 系统数据可视化

讲师和管理员都可以查看到系统的数据，登录成功即可看到系统的数据，例如学生的人数，讲师人数，昨日注册的学生人数，课程数量，教学设施数量，题库数量等系统数据；前端通过定时任务调用获取系统数据的接口来实现间隔刷新数据图表的数据，数据可视化通过使用百度 Echarts 框架的图表绘制功能实现，很直观显示相关的数据及其差异，提升了用户体验。

6.5 排课管理模块

6.5.1 排课任务列表

进入该界面自动请求后端获取排课任务的数据并填充显示在页面上，在每一行数据后面有删除，编辑按钮，可以执行对应的操作。页面上方有手动添加任务，下载导入模板，下载导入模板，导入数据，排课按钮。界面图如下所示：

2019-2020-1

手动添加

下载模板

从Excel导入

上传到服务器

排课

只能上传xls/xlsx文件,导入新任务后将清空原来的任务,请一次性将本学期的课程导入完毕

<input type="checkbox"/>	学期	年级	班级	课号	课名	课程性	讲师编号	讲师	学生人数	周学时	周数	固定	时间	操作
<input type="checkbox"/>	2019-2020-1	01	20200101	100001	高一语文必修1	01	10010	梁晓明	42	6	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100033	高一数学必修1	01	10012	李雪雷	37	6	20			<div>删除</div> <div>编辑</div>

图 6.6 排课任务列表效果图

6.5.2 Excel 导入任务

为了减少手动添加一个个操作带来的麻烦，系统提供通过 Excel 导入数据的方式，用户只需要下载系统提供的导入模板，根据说明将排课任务填写到 excel 文件中，再将文件上传到系统中，系统即可通过 easypoi 框架解析文件中的内容并将数据全部添加到数据库中，随后即可在排课任务页面中查看到本学期需要排课的课程。需要注意的是，填写数据一定要根据要求填写，否则会导致排课失败。导入的流程如下图所示：

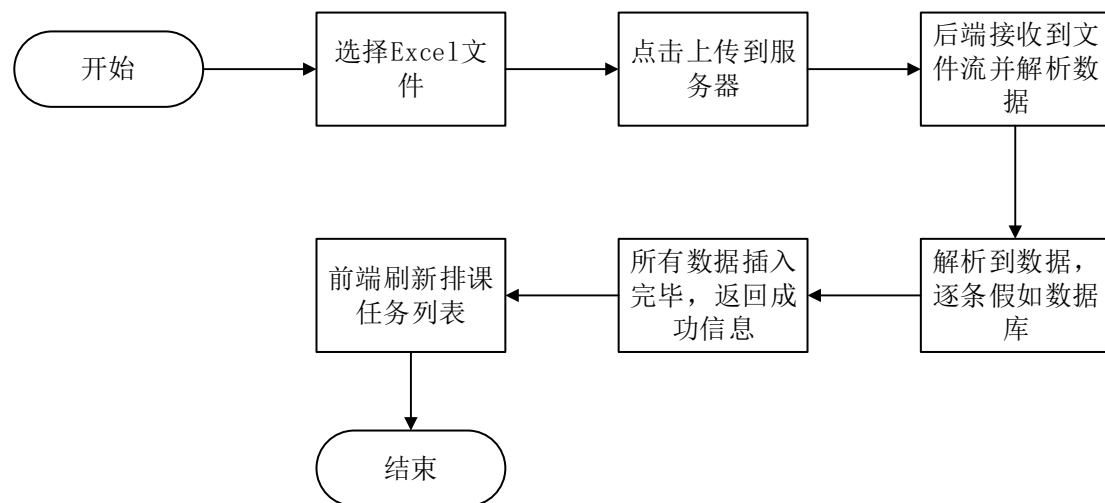


图 6.7 Excel 导入排课任务流程图

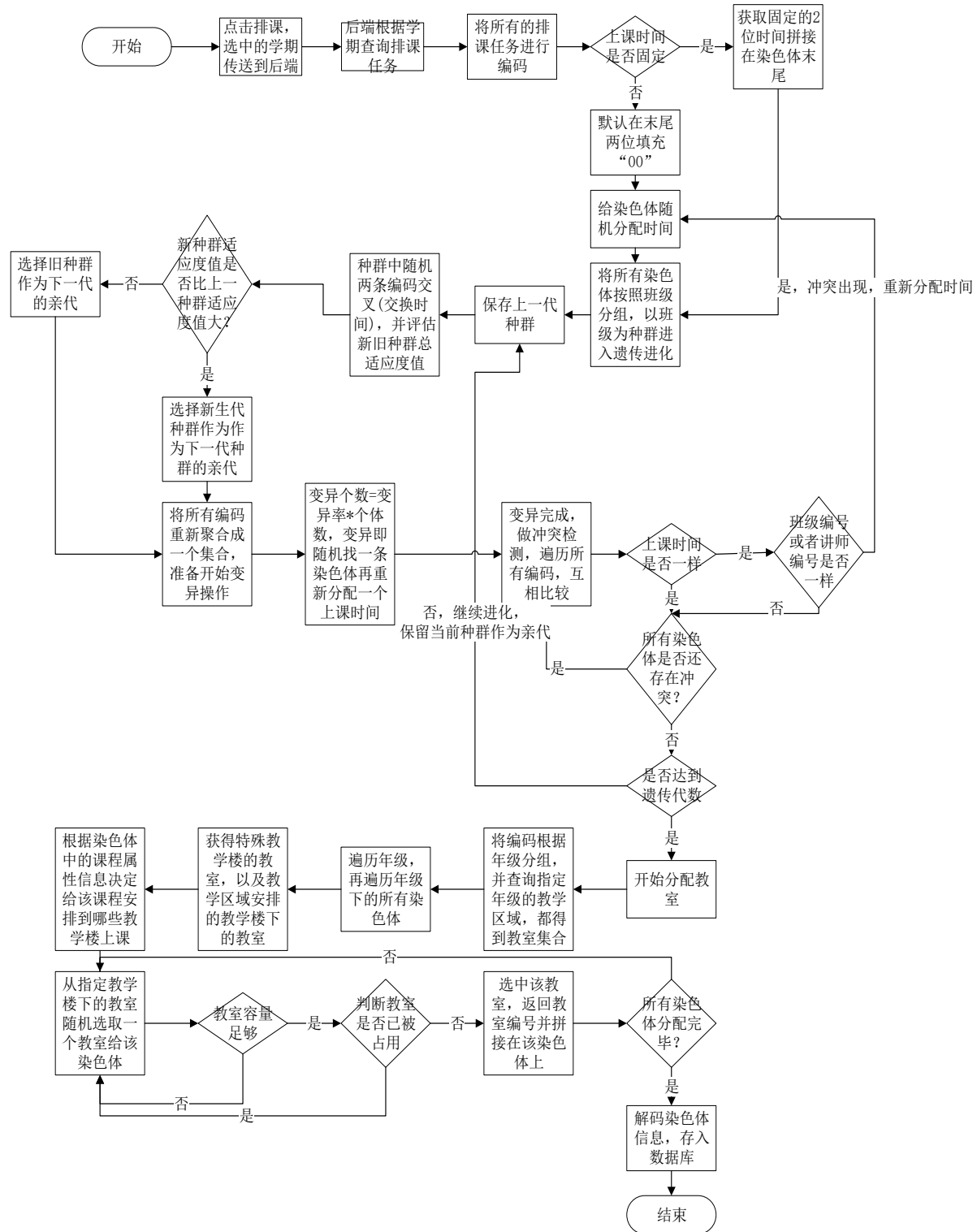
6.5.3 手动添加任务

由于每次导入新的上课任务都会清空之前所有的任务，因此用户要么一次性导入所有任务，要么就面临重新导入时全部清空的情况，这样就可能导致每次导入的时候都要重新打开 excel 文件将所有的任务都填写；系统提供了手动添加排课任务的方式，针对用户不想因为添加一两个任务就要重新导入文件的情况，减轻服务器压力。点击添加按钮后弹出表单，用户按要求填写弹窗中的内容后点击提交数据到后端，后端也会将数据插入到数据库中之后前端会刷新页面数据。

6.5.4 排课实现

所有的排课任务在导入之后都已经由系统后台解析存储到数据库中，排课只能一个个学期进行排课，即在每一次的任务列表中只能出现同一个学期的课程，当点击“排课”按钮时，前端将下拉选择所选中的学期值携带到后端，后端根据这个学期值去查询数据库中对应的任务，并将其放在集合中，随后循环将集合中的任务的相关字段拿出来拼接编码成字符串，得到未包含上课时间、上课教室的基因编码，编码规则如下：是否固定

+ 年 级 编 号 + 班 级 编 号 + 教 师 编 号 + 课 程 编 号 + 课 程 属 性 + 上 课 时 间， 例 如 10120200102100101000010100， 这 是 一 条 初 始 基 因 编 码， 随 后 将 不 固 定 上 课 时 间 的 编 码 全 部 拿 出 来， 给 这 些 编 码 随 机 生 成 一 个 01-25 编 号 的 上 课 时 间， 直 到 同 一 个 班 级 的 同 一 门 课 上 课 时 间 不 发 生 冲 突， 再 给 下 一 条 编 码 分 配 时 间， 直 到 所 有 的 编 码 都 分 配 上 时 间。 然 后 将 所 有 的 编 码 按 照 班 级 划 分， 整 个 班 级 的 编 码 作 为 一 个 种 群， 以 种 群 为 集 合 单 位， 将 集 合 中 的 任 意 两 条 编 码 随 机 交 换 其 上 课 时 间， 即 交 叉 操 作， 随 后 评 估 交 叉 前 后 的 种 群 适 应 度 值， 适 应 度 值 大 的 种 群 继 续 遗 传 下 去， 随 后 对 新 种 群 的 个 体， 即 基 因 编 码 进 行 变 异 操 作， 变 异 的 原 理 就 是 随 机 再 给 选 中 的 个 体 再 分 配 一 个 随 机 时 间， 随 后 进 行 冲 突 检 测， 防 止 同 一 老 师 在 同 一 时 间 同 时 上 同 一 门 课 的 情 况， 重 复 交 叉、 选 择、 变 异 步 骤， 直 到 算 法 到 达 终 止 条 件， 即 遗 传 代 数 结 束； 最 终 给 全 部 的 基 因 编 码 分 配 教 室， 即 在 编 码 后 加 上 教 室 编 号， 所 有 的 基 因 在 分 配 好 不 冲 突 的 教 室 之 后， 开 始 解 码 将 最 原 始 的 信 息 取 出 来， 存 放 到 数 据 库 中， 至 此 完 成 排 课。 排 课 的 流 程 图 如 下 图 所 示：



6.5.5 查看课表

查看课表页面, 上方有下拉菜单选择学期, 年级, 班级, 当选择学期和年级后会自动触发根据学期和年级查询班级的函数, 后端返回班级数据填充到班级下拉菜单中, 此时用户再选择班级, 再次出发根据班级查询课表的函数, 向后端发送请求, 返回上课任务的数据并根据前端的运算逻辑将数据规矩地显示在课程表上, 查看课表流程图以及课

表的效果如下图所示：

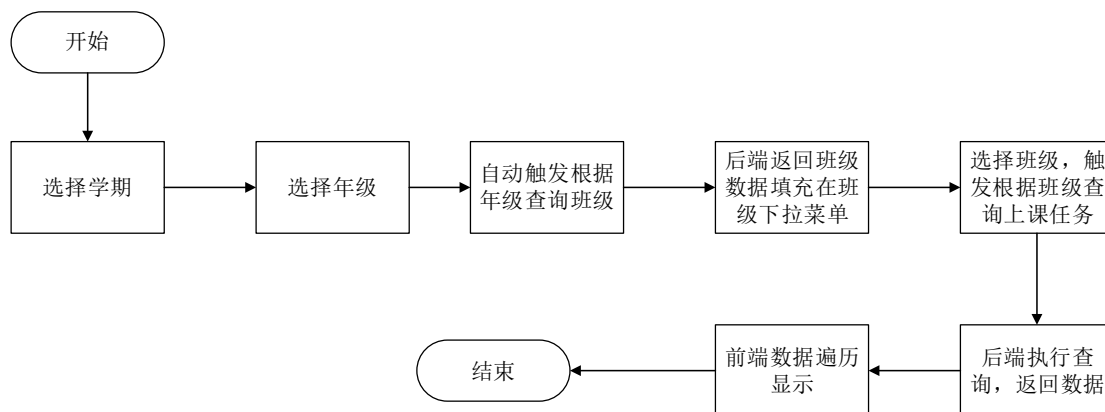


图 6.8 查看课表流程图

2019-2020-1	高一	20200101					
时间	周一	周二	周三	周四	周五		
第一节 07:20-8:55	张德良-高一物理1(01-303)	韩云-高一化学必修1(01-201)	梁晓明-高一语文必修1(01-305)	谭咏麟-高一生物必修1: 分子与细胞(01-103)	王小芳-高一英语必修1(01-401)		
第二节 9:10-10:45	王小芳-高一英语必修1(01-305)	李雷雪-高一数学必修1(01-204)	吴天盛-高一历史必修1(01-205)	李雷雪-高一数学必修1(01-504)	韩云-高一化学必修1(01-205)		
第三节 11:00-12:35	王杰-高一地理必修1(01-301)	江小珪-高一思想政治必修1(01-203)	谭咏麟-高一生物必修1: 分子与细胞(01-204)	李雷雪-高一数学必修1(01-301)	张德良-高一物理1(01-405)		
第四节 14:20-15:55	张德良-物理实验(08-103)	梁晓明-高一语文必修1(01-401)	张杰-体育课(12-203)	江小珪-高一思想政治必修1(01-202)	王杰-高一地理必修1(01-103)		
第五节 16:10-17:45	-	吴天盛-高一历史必修1(01-503)	张麒麟-化学实验(08-302)	梁晓明-高一语文必修1(01-303)	王小芳-高一英语必修1(01-104)		

图 6.9 课程表效果图

6.6 课程管理

6.6.1教材管理

教材管理模块主要管理的是学校平时上课所使用的教材资料，在排课之前设定排课任务时，使用教材的名字以及编号作为开设的课程，在这个菜单下实现的功能有教材的添加，删除以及修改，还可以在海量教材中通过教材名称关键字模糊搜索快速找到相关的教材。

点击添加教材按钮，弹出表单的同时会自动向后端发送一个请求来获取教材的编号，不需要用户手动输入，以防出现编号冲突导致排课时出现错误；在新增表单中需要填写教材的名称，属性，也就是排课任务中用到的课程属性，出版社和备注信息选填，填写完毕点击提交即可完成教材的添加。添加教材流程如下图所示：

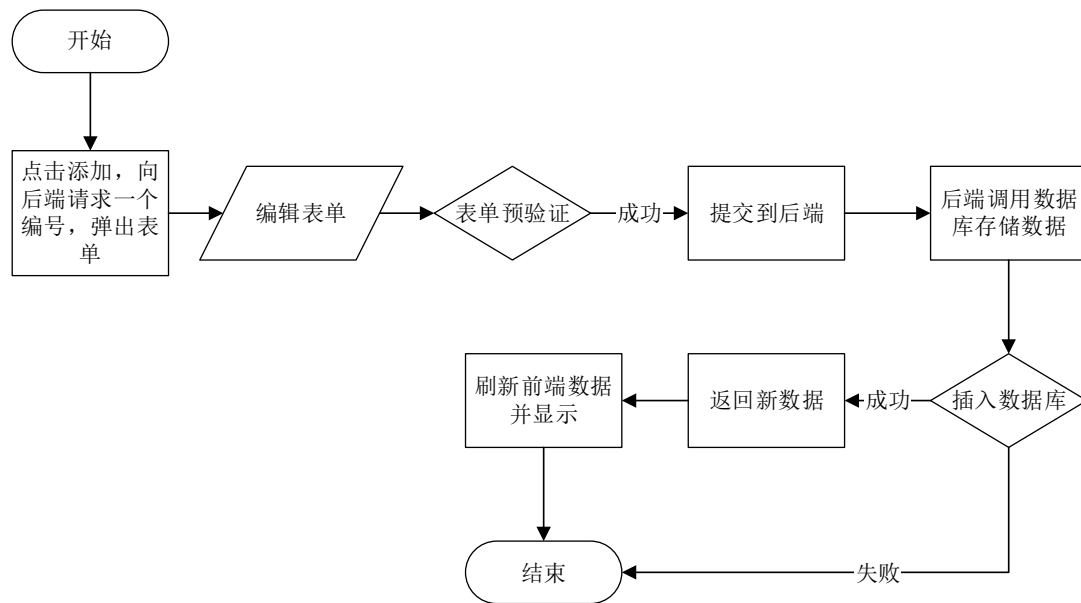


图 6.10 添加教材流程图

因为在前端教材列表显示之前已经从后端获取到了这些教材信息的数据，因此在点击编辑按钮或者删除按钮时都可以根据已有的信息去操作，编辑完新的内容后提交到后端时会根据提交信息中的 id 更新该实体信息；删除则是根据 id 直接去删除该记录。

6.6.2 网课类别管理

网课类别分为一级类别和二级类别，在二级分类下面才有课程，每个课程下还有一个或者多个网课视频。点击添加一级类别弹出表单，填写类别编号和类别名称，提交即可；如果是添加二级类别，在点击添加二级类别按钮后，会触发查询所有的一级类别存放在表单中的一级类别下拉选项中，需要选择一级类别之后才可以填写二级类别的编号和名称，点击提交即可。

6.6.3 网课管理

添加网课时，只有选择了一级分类和二级分类之后才可以添加课程，添加课程时需要先上传一张图片作为课程的封面，随后在表单中填写课程名称与课程简介，点击提交完成课程的添加。课程添加的流程图如下图所示：

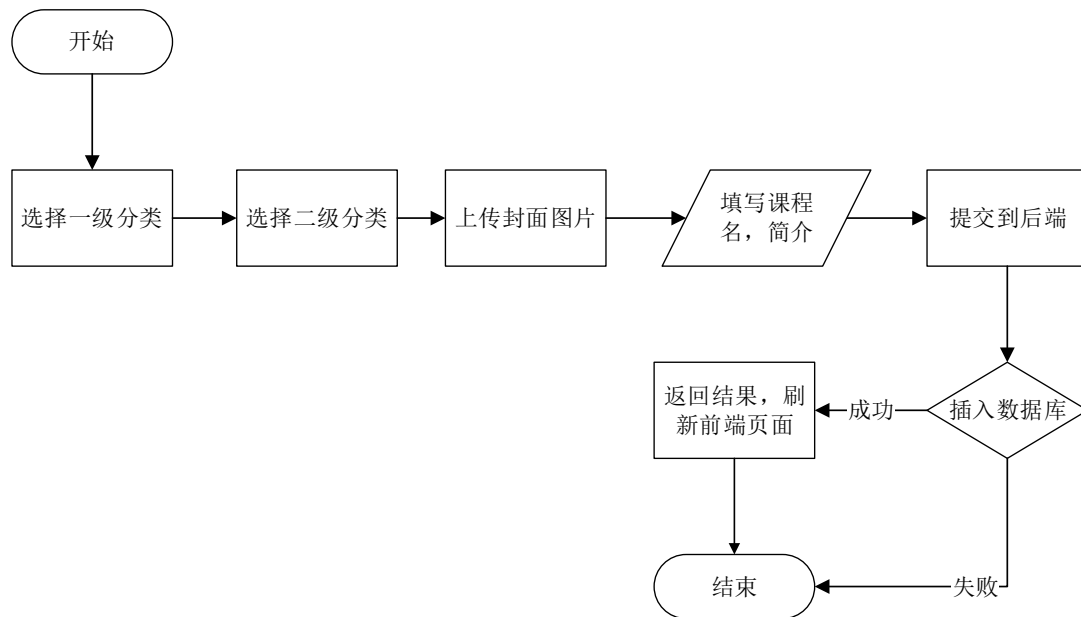


图 6.11 添加网课流程图

上传网课视频到指定课程下面，点击上传视频弹出表单，首先选择到目标课程，再选择本地视频文件后方可将视频上传到对应的课程下。上传网课视频的流程图如下：

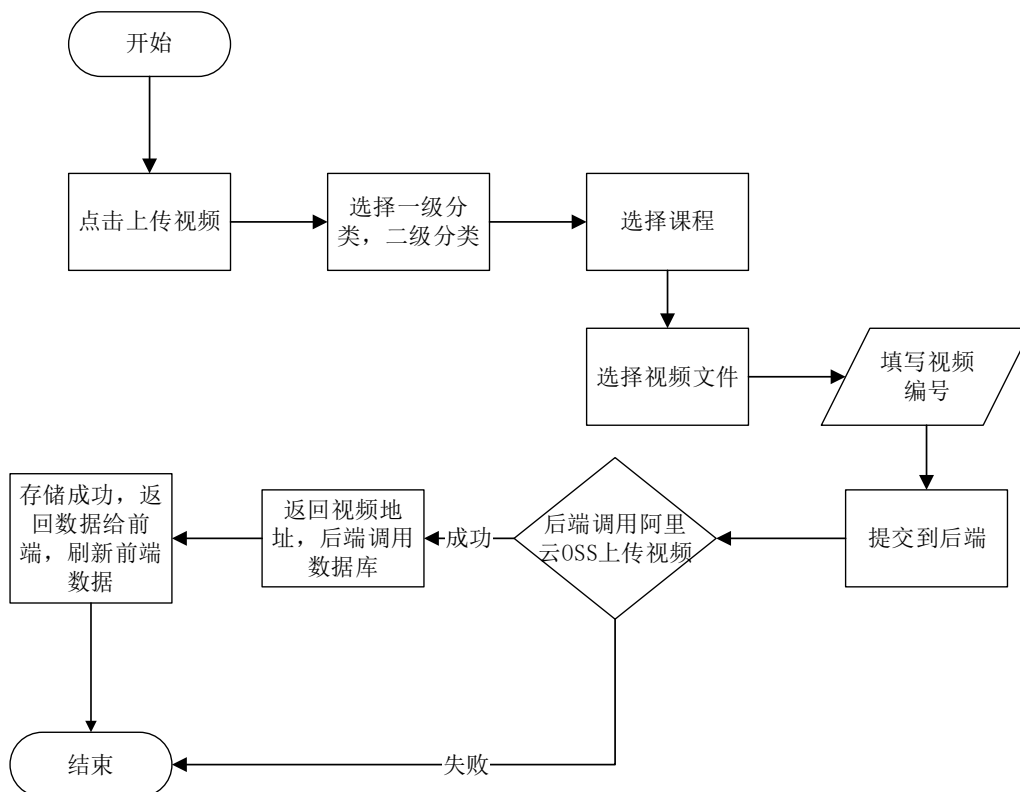


图 6.12 上传网课视频流程图

6.7 讲师管理

在讲师管理下是所有讲师的菜单，在该页面下可以执行添加讲师，删除讲师，编辑讲师操作。添加讲师也即前面所述的讲师注册。

在讲师数量较多的情况下，可以通过讲师真名中的关键字查询讲师，在搜索框中输入关键字，点击搜索按钮，关键字提交到后端并匹配数据库，后端将查询结果返回前端并显示，即可看到搜索结果。

点击删除按钮即可实现讲师的删除，删除逻辑是通过讲师信息在前端存储的数据中的 id 提交到后端，后端根据讲师的 id 逻辑删除在数据库中的记录，逻辑删除并不是真正将数据从数据库移除，只是将数据库中的一个字段修改其值，使其在正常查询下查询不到。

点击编辑讲师，从前端获得该讲师的信息并显示在弹出的表单上，用户将新的信息编辑好在表单中，可编辑的有讲师昵称，姓名，职称，任教科目，手机，邮件等信息，编辑完成并通过了表单验证，点击提交即可将数据传送到后端执行更新操作，完成更新后会重新刷新数据。编辑讲师的流程图如下所示：

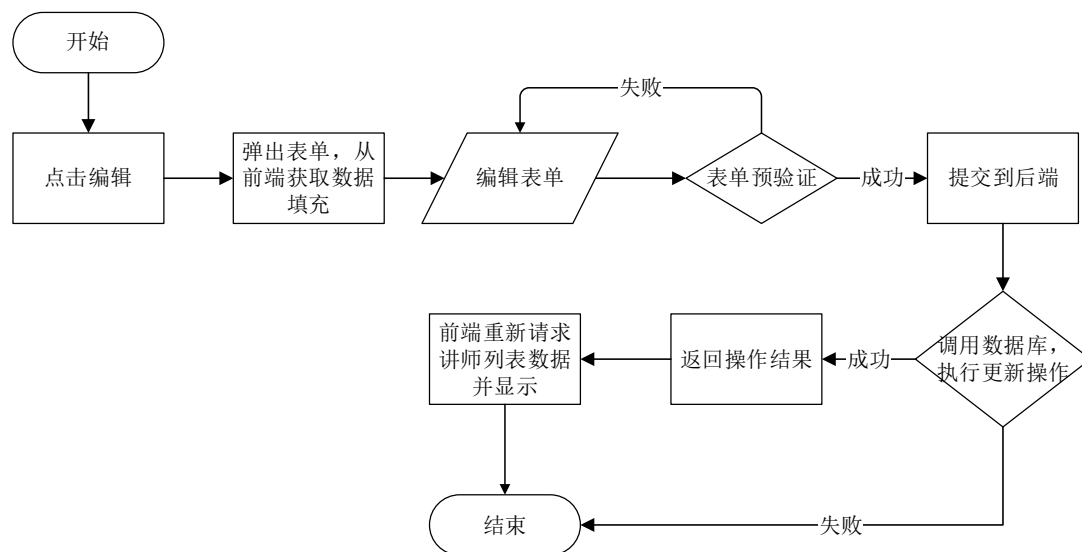


图 6.13 编辑讲师流程图

6.8 班级管理

在班级管理下面，点击所有班级菜单即可在加载该页面时自动请求后端所有班级接口并显示班级列表，每一行内容包含所属年级，班级编号，班级名称，班主任，学生人数，以及后面的编辑和删除按钮。在页面上方有下拉菜单通过选择年级来查看该年级下的班级，当选择年级时，会触发根据年级编号查询班级，向后端发起请求，返回的数据展示在页面上。

页面上方还有添加班级按钮，点击之后弹出添加班级表单的同时会触发查询所有讲

师函数，将查询到的讲师填充在班主任选择的下拉菜单中，用户选择年级，选择班主任，再将班级编号，班级名称填写完毕，点击提交到后端，后端调用数据库将前端提交的数据插入数据库，插入成功后返回成功代码给前端，前端得到指示后重新获取所有班级信息并显示在页面上。以下为添加班级的流程图：

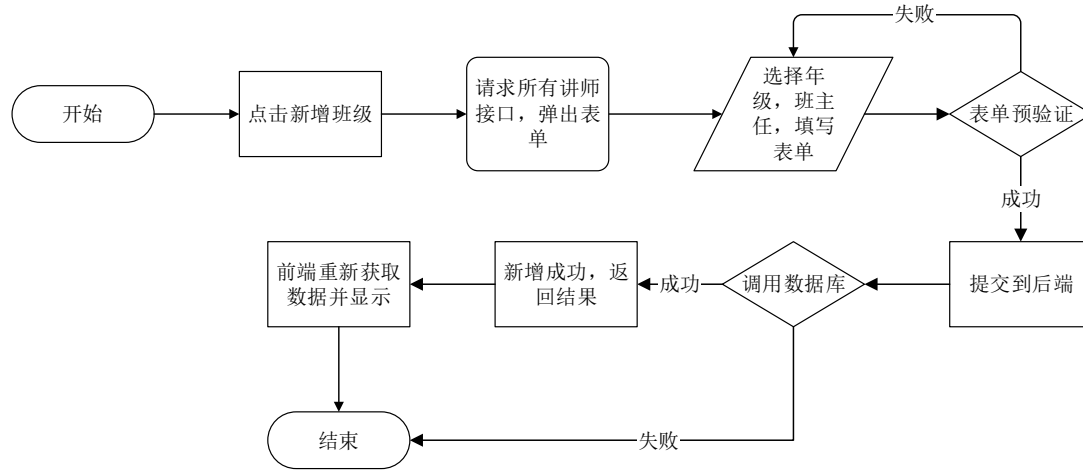


图 6.14 新增班级流程图

6.9 学生管理

学生管理模块的功能有：搜索学生，根据班级查看学生，编辑学生，删除学生。

搜索学生，在搜索条中输入学生姓名的关键字，点击搜索，前端携带关键字提交到后端模糊查询数据库并返回匹配的学生数据经过前端计算显示在页面上。

以班级为单位显示学生，页面上方有两个下拉菜单：年级和班级，选择年级后触发查询该年级下的班级，再下拉选择班级时将选中的班级传输到后端根据班级查询学生，返回前端并显示在页面上。

删除学生，本地存储有学生的信息，每个学生的唯一标识是 id 号，点击删除按钮时，请求后端删除接口并将 id 携带过去，后端通过 id 删除学生记录，删除成功后前端重新调用获取所有学生的接口并展示数据。

编辑学生与编辑讲师的流程类似，其流程图如下所示：

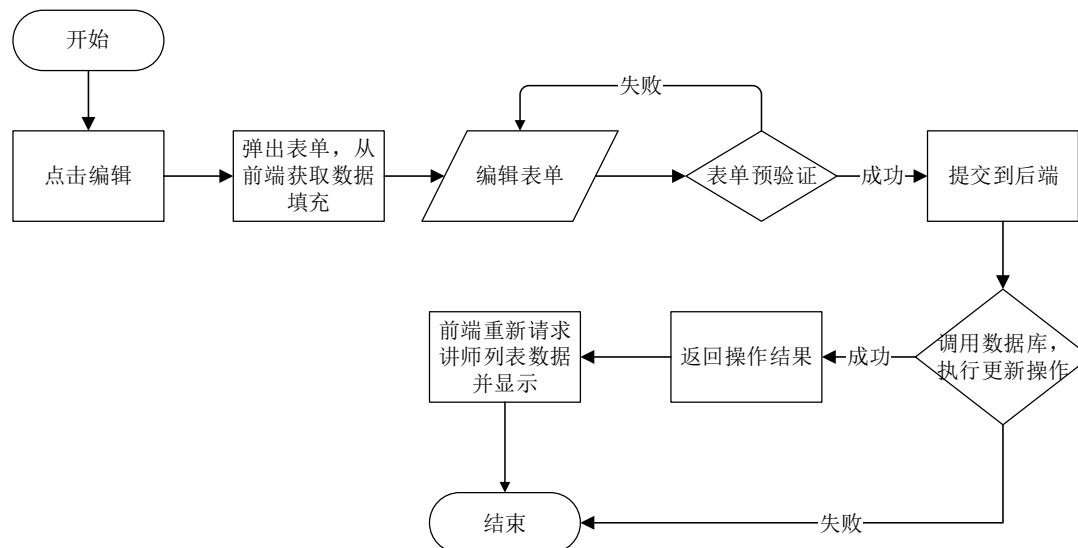


图 6.15 编辑学生流程图

6.10 教学资料管理

6.10.1 题库管理

题库管理中可以分题目类型进行管理，页面上方可以通过下拉列表选择题目类别来分类查看题库信息。

新增题目类型，只需要在弹出的表单中填写类型的名称并点击提交即可完成类型的添加，随后可以在下拉列表中看到。

添加题目，在添加表单弹出来之前请求后端接口查询所有的题目类别，在弹出框中需要下拉选择类别才可以将即将添加的题目附属到该类别下，再填写班级编号用于将题目设定为指定班级可以查看，默认所有班级可以查看，还需要填写题目的标题每一个选项的值，题目分值，是否多选以及题目的答案。以下为添加流程图：

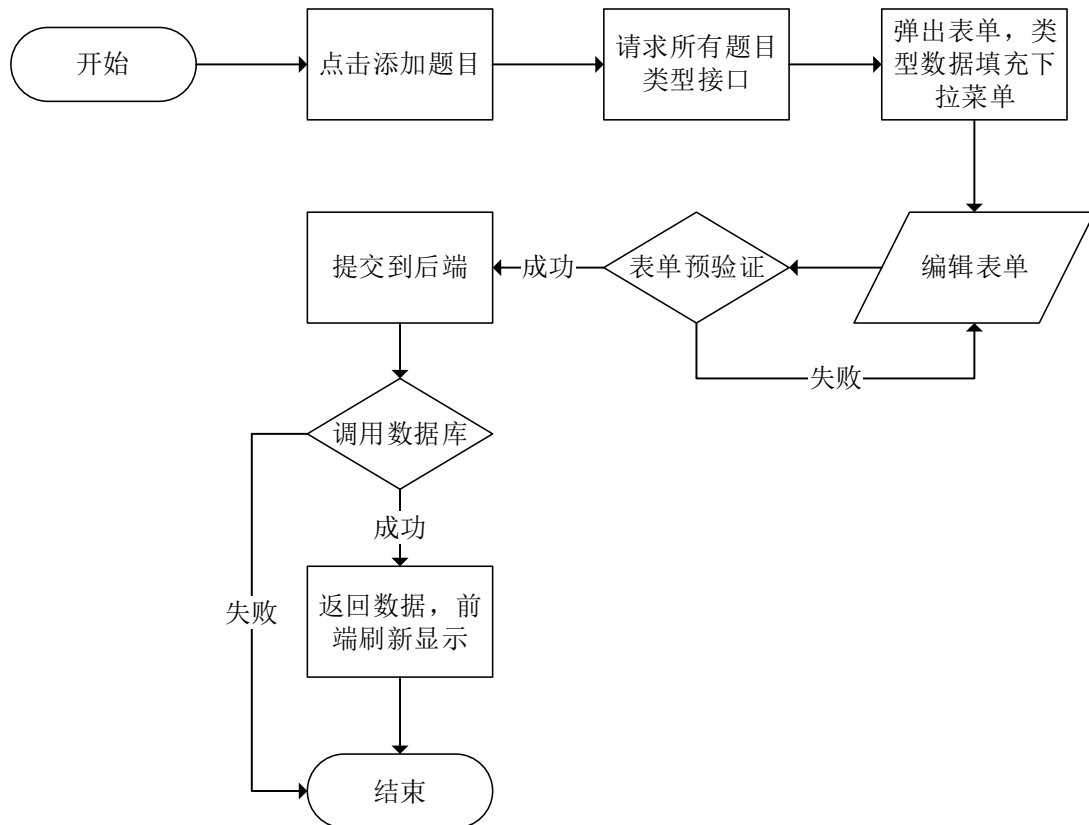


图 6.16 新增题目流程图

6.10.2 学习文档

学习文档主要是一些 office 文档, 讲师将文档上传到指定班级后, 该班级可以查看, 文档作为学习资料或者作业发布的载体, 在教学工作中扮演者重要角色。用户点击新增文档时弹出表单, 选择需要上传的文档, 每次只能上传一个, 选择好文档文件后就会自动将文件上传给后端, 后端再调用文件上传方法将文件上传到云端并将文件在云端的访问地址返回到前端, 用户在这个过程中也需要填写需要推送到的班级编号, 文档的简介等信息, 最后点击提交即可将用户编写的信息以及上传返回的 url 地址一起提交到后端存入数据库, 存储成功后前端自动刷新数据并显示。以下为上传文档的流程图:

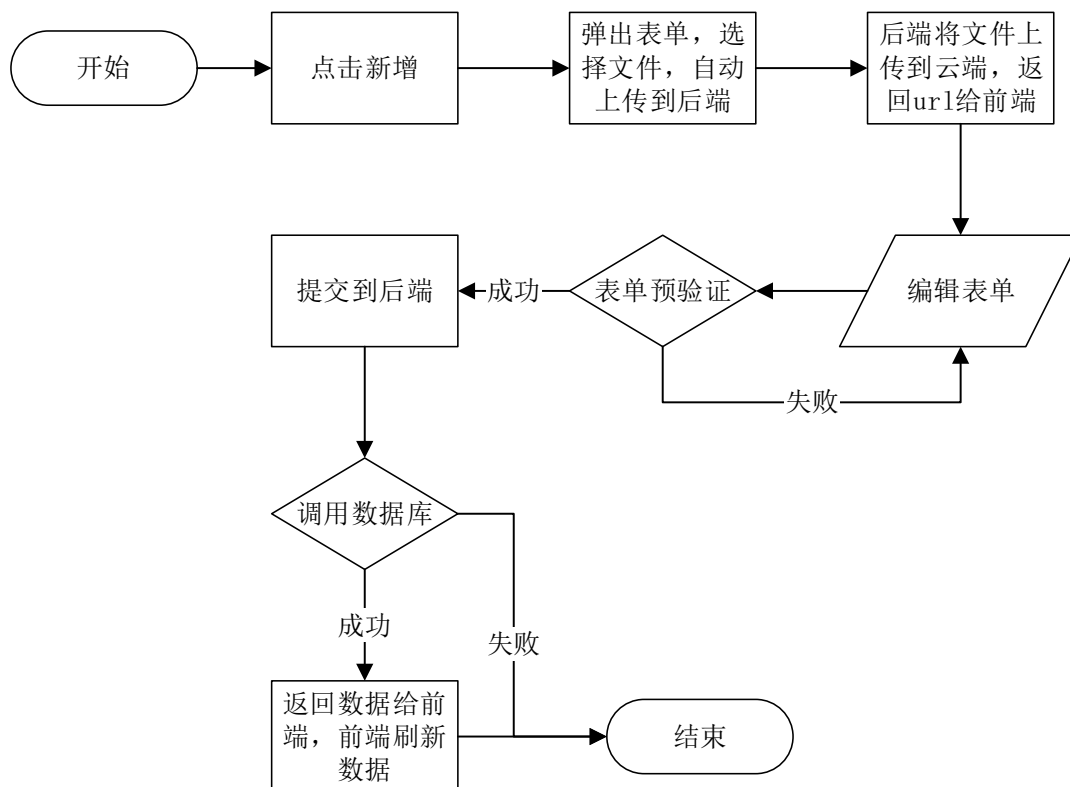


图 6.17 新增文档流程图

6.11 教学设施管理

6.11.1 教学楼管理

该菜单下可以看到学校所有的教学楼信息，教学楼编号、名称及其所属区域，还可以新增教学楼来满足教学过程中的需求，此外还可以删除、更新教学楼信息。

添加教学楼时，只需要在添加表单中填写教学楼编号，名称和所属区域后点击提交即可添加完成，效果图参考下图：

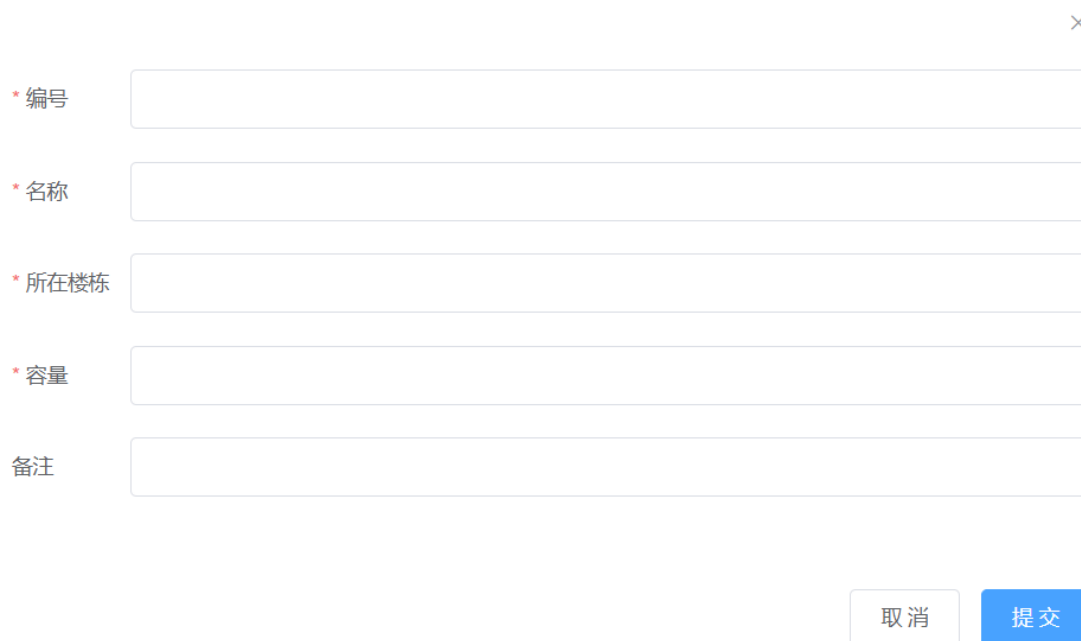
编号	<input type="text" value="11"/>
* 名称	<input type="text" value="测试楼"/>
* 所在区域	<input type="text" value="东校区"/>

图 6.18 添加教学楼界面图

6.11.2 教室管理

该菜单下显示全校所有的教室信息，教室所在教学楼，教室编号，教室名称，容量等。可以对教室执行更新、删除操作。

需要添加教室时，点击上方添加按钮即可在表单中填写教室编号，名称，所在楼栋，容量等信息，切记要将教室编号，所在楼栋填写正确以及编号不可以有重复，否则可能导致排课过程出现错误。添加教室页面如下图所示：



The image shows a web form for adding a classroom. It has a title bar with a close button (X). The form contains five input fields: '编号' (Number) with a red asterisk, '名称' (Name) with a red asterisk, '所在楼栋' (Building) with a red asterisk, '容量' (Capacity) with a red asterisk, and '备注' (Remarks). At the bottom right, there are two buttons: '取消' (Cancel) and '提交' (Submit).

图 6.19 添加教室界面图

6.11.3 教学区域管理

教学区域主要用于设置每个年级上课所在的教学楼，如果某一栋教学楼的教室无法满足一个年级上课了，那么可以新增教学区域设置的记录，在新增表单中下拉选择年级，选择教学楼添加即可，下拉菜单中的数据在点击新增按钮时已经向后端请求获取到并填充到了下拉菜单中，点击提交后，系统后台会进行判断，若教学区域已存在，那么添加失败。设置效果图如下：

设置教学区域

请选择年级

请选择教学楼

取消

提交

图 6.20 添加教学区域界面效果图

7 系统测试

7.1 学生注册测试

表 7.1 注册测试用例表

操作	预期结果	测试结果
表单中输入必填信息，点击注册	注册成功并且跳转到登录页面	结果与预期符合

注册页面如图所示：

学生注册

registerTest

.....

.....

我是测试用户

高一

2020014949

创建学号

13677731234

test@qq.com

桂电

注册

已有帐号

图 7.1 注册页面效果图

注册成功出现弹窗提示，效果图如下：



图 7.2 注册成功图

7.2 登录测试

管理员与讲师共用同一个登录页面，在输入账号密码之后选择登录类型即可。

表 7.2 用户登录测试用例表

操作	预期结果	测试结果
表单中输入正确的账号密码，点击登录	登录成功，跳转到登录成功页面	结果与预期符合

登录页面效果图如下：



图 7.3 讲师，管理员登录界面图

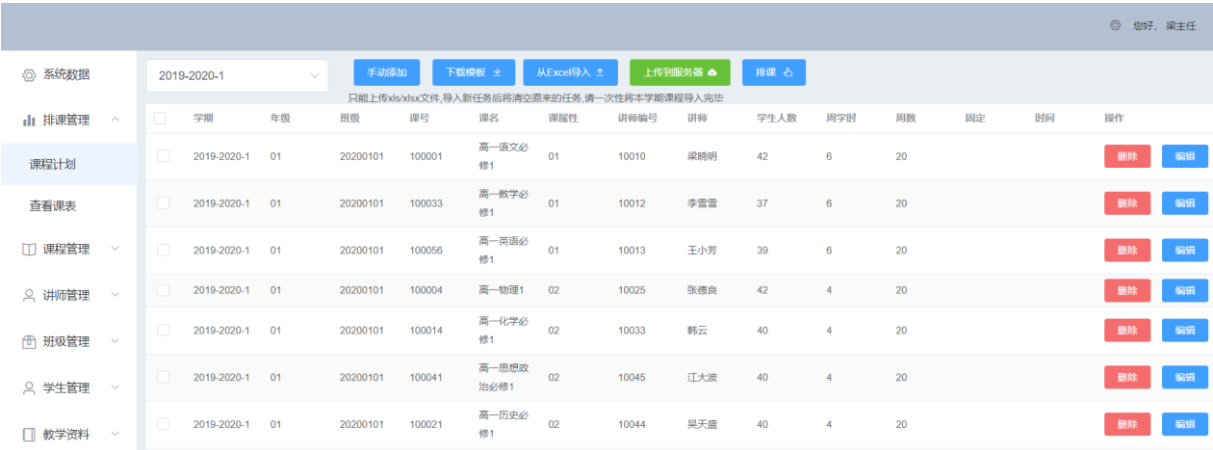


图 7.4 登录成功图

7.3 排课测试

在排课之前，首先确保要有待排课的任务，否则无法排课。

表 7.3 排课测试用例表

操作	预期结果	测试结果
在确认正确导入排课任务，确保排课任务可以显示后，可以点击排课按钮开始排课	排课成功跳转到查看课表页面，并且可以根据不同的班级查询到课表	结果与预期符合

排课成功效果图如下所示：

2019-2020-1	高一	20200102			
时间	周一	周二	周三	周四	周五
第一节 07:20-8:55	王小芳-高一英语必修1(01-203)	王小芳-高一英语必修1(01-405)	王杰-高一地理必修1(01-503)	李雪雪-高一数学必修1(01-204)	韩云-高一化学必修1(01-403)
第二节 9:10-10:45	吴天盛-高一历史必修1(01-303)	-	李雪雪-高一数学必修1(01-305)	张德良-高一物理1(01-205)	梁晓明-高一语文必修1(01-504)
第三节 11:00-12:35	-	谭咏麟-高一生物必修1：分子与细胞(01-505)	梁晓明-高一语文必修1(01-301)	江小陆-高一思想政治必修1(01-102)	李雪雪-高一数学必修1(01-303)
第四节 14:20-15:55	王杰-高一地理必修1(01-205)	张德良-物理实验(08-104)	-	张杰-体育课(12-203)	梁晓明-高一语文必修1(01-101)
第五节 16:10-17:45	-	王小芳-高一英语必修1(01-303)	吴天盛-高一历史必修1(01-104)	张麒麟-化学实验(08-202)	江小陆-高一思想政治必修1(01-502)

图 7.5 排课完毕结果图 1

7.4 导入排课任务测试

表 7.4 导入排课任务测试用例

操作	预期结果	测试结果
将任务按照模板要求填写完成后，选择该文件并上传	上传成功后会刷新显示排课任务列表	结果与预期符合

导入课程任务之后系统会自动更新排课任务的列表，排课任务列表效果图如下图所示：

2019-2020-1

手动添加

下载模板

从Excel导入

上传到服务器

排课

只能上传xls/xlsx文件,导入新任务后将清空原来的任务,请一次性将本学期课程导入完毕

<input type="checkbox"/>	学期	年级	班级	课号	课名	课程性	讲师编号	讲师	学生人数	周学时	周数	固定	时间	操作
<input type="checkbox"/>	2019-2020-1	01	20200101	100001	高一语文必修1	01	10010	梁晓明	42	6	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100033	高一数学必修1	01	10012	李雪雪	37	6	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100056	高一英语必修1	01	10013	王小芳	39	6	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100004	高一物理1	02	10025	张德良	42	4	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100014	高一化学必修1	02	10033	韩云	40	4	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100041	高一思想政治必修1	02	10045	江大波	40	4	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100021	高一历史必修1	02	10044	吴天盛	40	4	20			<div>删除</div> <div>编辑</div>
<input type="checkbox"/>	2019-2020-1	01	20200101	100007	高一地理必修1	02	10043	王杰	40	4	20			<div>删除</div> <div>编辑</div>

图 7.6 排课任务列表图

8 结论与展望

8.1 结论

经过这段时间各方面的努力，在课程编排问题的算法选取上，经过对比也选择了当下解决组合约束问题使用较为广泛的遗传算法。经过对遗传算法的一段时间的了解以及相关资料的查阅，也成功将遗传算法在代码实现上体现出来，完成了课程安排的功能。此外，系统还可以辅助开展教育教学工作，比如实现了讲师管理，学生管理，班级管理，学习资料管理，作业发布等功能。

系统的优点，通过编程实现课程编排功能，借助于计算机强大快速的计算能力，即使在众多课程任务的情况下，也可以在极短的时间内完成课程的编排，这也满足了对每周课程安排不同的学校教学模式生态，有该需求只需每周都执行一次排课即可。许多教学辅助功能，也给师生带来了很大的便利，方便学校资源的管理。

当然也存在一些缺点，在实现排课的过程中，由于更注重的是技术上的实现，更关注如何利用编程语言去模拟进化的过程，因此在理论知识上并没有做太深入的理解。需

要注意的是，遗传算法在实现过程中主要的过程就是对种群进行选择、交叉、变异，随后重复以上几步操作，直到算法到达终止条件，比如到达了设定的遗传代数亦或者是其它的一些异常出现；遗传算法在使用上一定要注意的就是适应度函数的设计，变异率，交叉率，由于这是多个参数组合，因此在调参上肯定需要花费很多的时间，即使经过多次调参，可能得到的结果也不一定是个人满意的。后两者通常设置得比较低，而尽可能通过变换种群的遗传代数去搜索全局结果；倘若变异率设置过高，那么算法就过快收敛；如果设定过低，有可能在算法结束时还未找到较优的解，因此，在参数的设置需要不断测试，测试次数有限也导致本次系统在排课算法上并不是很优秀，比如尽量让主要课程往上午安排，但还是出现了安排在下午的情况。

在实现过程中也发现了算法一些严重的问题，在给染色体分配时间的时候，由于一直处于递归调用，因而在课程任务很多的时候，容易出现堆栈溢出异常导致程序退出。通过修改分配时间的函数，消除递归也解决了这个问题。

8.2 展望

这个系统实现出来还是存在很多缺点，就例如安排的课程上，主要课程与次要课程确实对其尽量限制在某一个时间段上课，但结果还是有不少的出入，因而在后续应该抽时间将适应度函数进行优化，从而使得排课效果更优秀。

此外，预期的时候希望排课的时候还可以根据不同的周去排课，从而通过选择不同的周查看该周的课表，但由于能力有限并未在系统中完成该功能；但是起码借助其高效快速的运算能力，也可以每周都进行一次排课从而实现每周的课表都不一样。

谢 辞

四年的大学时光即将结束，很感谢这四年以来学校给我提供的平台以及各方面的学习资源，让我受益匪浅，也依靠学校的影响力找到了自己心仪的工作。

在经历了三个多月的努力，也将毕业设计以及论文完成了，整个过程里很感谢我的指导老师潘海玉老师，从开始选题到论文完成，给了我许多的帮助，在算法选择上给了我很多建议，在实现软件系统以及撰写毕业论文的过程中也给了我不少的指导，使我在在这个过程中少走了很多弯路；另外也很感激王学长在毕设过程中的技术指导，让我在技术的使用上也没有遇到太多难以解决的问题。

在论文编写过程中，我的同学也给了不少的帮助，让我在调整格式上省下了不少的时间。在这四年里，在学习文化知识过程中或者是学习技术过程中，都得到了身边很多同学的帮助，能有今天或多或少都有他们的功劳，在即将离别，所剩无几的日子里，我应该好好珍惜这些时光，用一些行动去感激那些曾给予我帮助的老师，同学们。

能走到今天，更离不开我的父母含辛茹苦的栽培与默默支持，在此也向我的父母表达深深的敬意。

即将步入社会的我，能够拥有一段大学时光，也是我人生中很荣幸的一件事情，在以后的人生路中，回想起自己在大学里的生活，也肯定会有一股深深的怀念之情。再次感谢一路走来身边出现的每一个人。

参考文献

- [1] 钱海军.基于遗传算法的开放教育排课系统研究[D].广东:广东技术师范学院,2018.
- [2] 王卫红,李文琼.基于改进遗传算法的高中走班制排课算法[J].浙江工业大学学报, 2016, 44(6):601-607.
- [3] 郝晓鹏.基于遗传算法的排课选课系统设计与实现[D].北京:北京工业大学,2018.
- [4] 李茜.排课问题的算法研究[D].湖北:长江大学,2016.
- [5] 胡钜琿.遗传-蚁群混合算法在排课系统中的研究与应用[D].西安:西安科技大学,2019.
- [6] 张义广.高等院校排课系统的设计与实现[D].江西:南昌大学,2018.
- [7] 杨开振,周吉文,梁华辉,谭茂华,著.Java EE 互联网轻量级框架整合开发[M].北京:电子工业出版社,2017.
- [8] [美] Lee Jacobson & Burak Kanber. JAVA 遗传算法编程[M].王海鹏,译.北京:人民邮电出版社,2016.
- [9] 吴松慧.基于遗传算法的智能排课系统设计与实现[D].苏州:苏州大学,2016.
- [10] Chengyang Li. Genetic Algorithm to Solve the Problem Arranging Course[C]. 信息化与工程国际学会.Proceedings of 2016 6th International Conference on Machinery,Materials,Environment,Biotechnology and Computer(MMEBC 2016).信息化与工程国际学会:计算机科学与电子技术国际学会(Computer Science and Electronic Technology International Society),2016:1039-1042.
- [11] 孙光民,赵莹帝,周青昱.基于改进遗传算法的新高考体制优化排课系统[J].电子世界,2020(04):74-78.
- [12] 王华鑫.走班排课算法的研究与设计[D].华中师范大学,2019.
- [13] A. Araisa Mahiba,C. Anand Deva Durai. Genetic Algorithm with Search Bank Strategies for University Course Timetabling Problem[J]. Elsevier Ltd,2012,38.
- [14] 王艺霏,代红.基于遗传算法的排课系统的设计与实现[J].电脑迷,2018(08):83.