

1 详细设计

1.1 开发工具

系统开发使用的工具及其版本如表 1.1 所示。

表 1.1 开发工具及版本

开发工具	版本	开发工具	版本
IDEA	2019.3	WebStorm	2021.3.2
vue	2.6.14	Element-UI	2.15.8
vue/cli-service	5.0.4	MyBatis-Plus	3.4.3
npm	8.9.0	node	16.14.2
java	1.8	springboot	2.6.6
Druid	1.2.6		

1.2 数据库设计

繁星之辰教学辅助系统中采用 MySQL 数据库管理系统来统一组织和管理系统用到的数据。根据需求分析可知系统中的实体主要有：管理员、用户、课程、通知、课件、讨论、回复、章节、实验、公告、报告和考试，其中考试又关联到：试题、选择题、选项、填空题、空项、判断题、简答题、考试记录、答卷和答案项等实体。

在这些实体中，有部分实体可以通过设置外键来关联其他实体，例如给课程表中设置用户 ID 来标识课程的创建者，给答卷表设置用户 ID 来标识答卷学生，以及可以给课件表、讨论表、章节表、考试表等设置课程 ID 来标识所属的课程，因为课程和教师是一一对应的，这些表就不再需要重复设置创建者 ID 了。此类外键如图 1.1 所示。

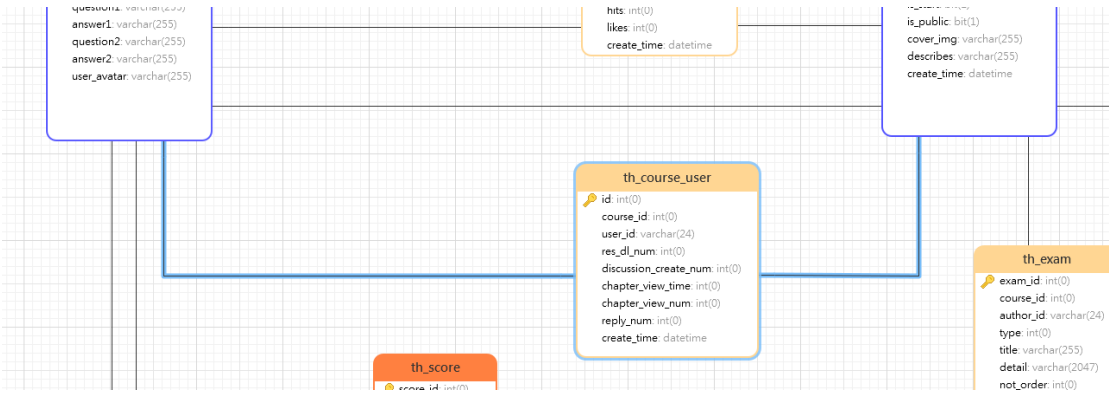


图 1.1 th_course_user 表两侧粗线段表示物理外键

系统中也有部分实体，在逻辑上与其他多个实体存在关联，但无法通过设置

综上所述设计出系统的 E-R 图，如图 1.2 所示。

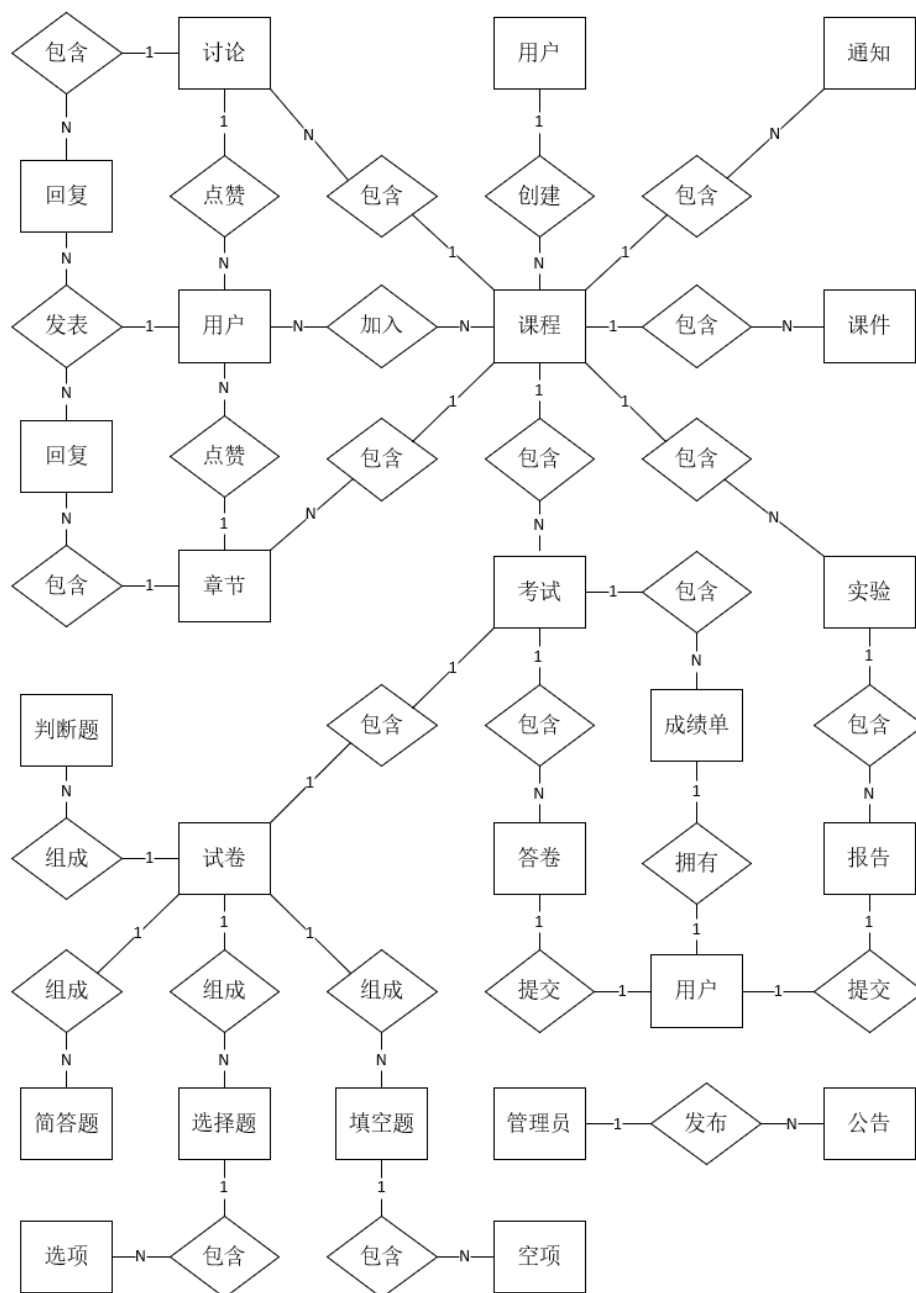


图 1.2 系统 E-R 图

1.2.1 数据库概念设计

如前文所述，系统中的主要实体有：用户、课程、讨论、回复、考试、答卷、成绩单等。

系统中用户表是学生表和教师表的综合，其所含属性有：用户 ID、密码、用户名、电话、邮箱、密保问题 1、密保问题 2、密保答案 1、密保答案 2、用户头像 URL。用户的实体图如图 1.3 所示。

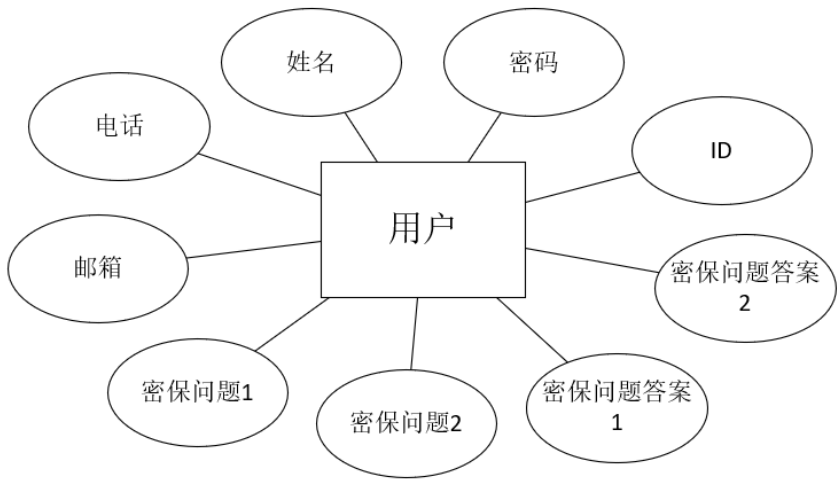


图 1.3 用户实体图

系统中课程表所含属性有：课程 ID、课程名、创建用户 ID、是否公开、是否激活、封面图 URL、课程简介、创建时间。课程的实体图如图 1.4 所示。

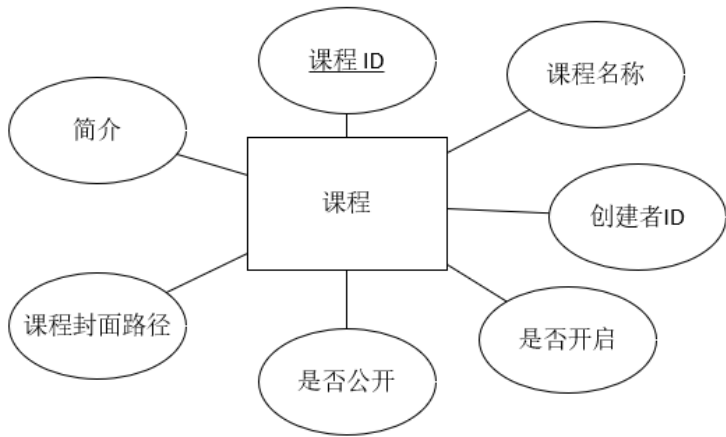


图 1.4 课程实体图

系统中讨论表所含属性有：讨论 ID、课程 ID、发布者 ID、标题、内容、点击量、点赞数、创建时间。讨论的实体图如图 1.5 所示。

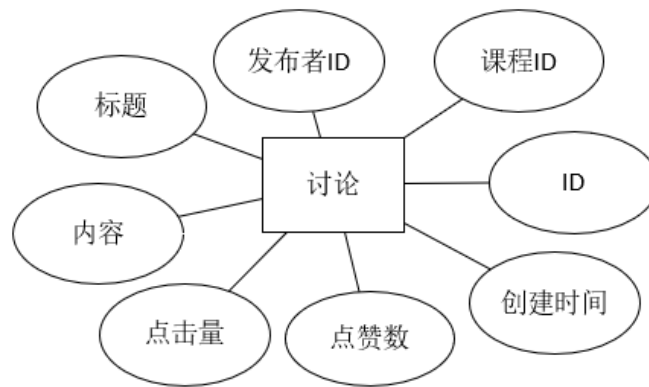


图 1.5 讨论实体图

系统中回复表所含属性有：回复 ID、所在讨论 ID、所在章节 ID、发表用户 ID、回复用户 ID、内容、创建时间。回复的实体图如图 1.6 所示。

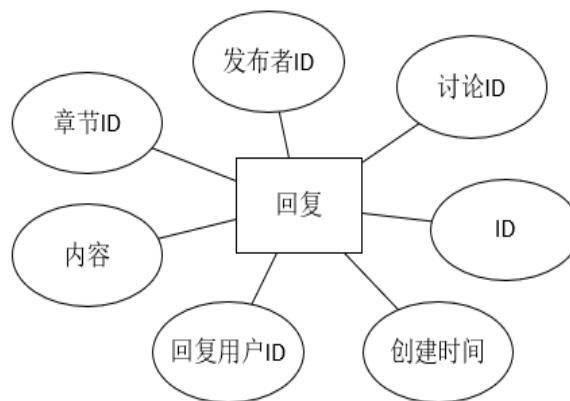


图 1.6 回复实体图

系统中考试表所含属性有：考试 ID、所在课程 ID、发布者 ID、类型、标题、详情、是否无序出题、开始日期、结束日期、创建日期。考试实体图如图 1.7 所示。

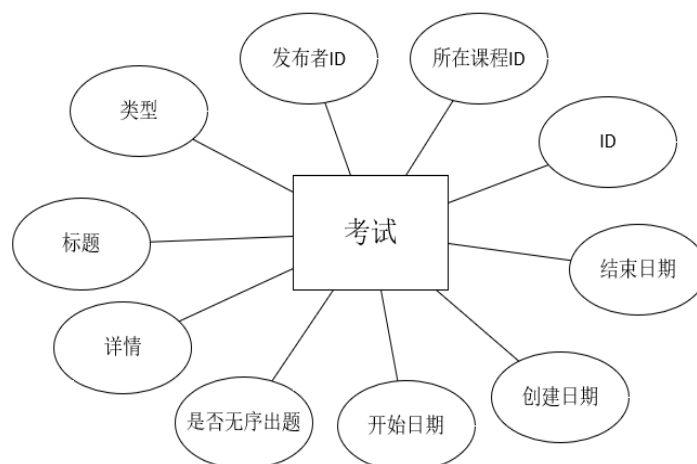


图 1.7 考试实体图

系统中答卷表所含属性有：答卷 ID、对应考试 ID、题目类型、题目 ID、题目序号、答卷者 ID、答案序号、答案内容、得分。答卷的实体图如图 1.8 所示。

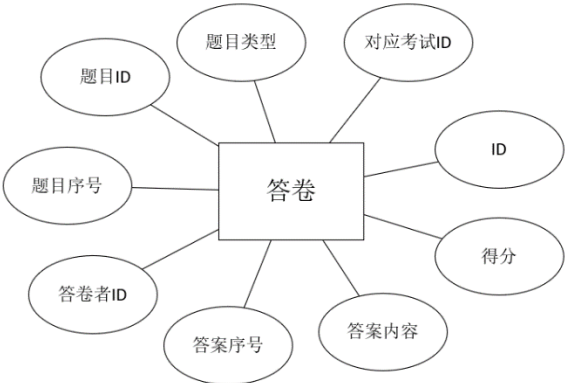


图 1.8 答卷实体图

系统中成绩单表所含属性有：成绩单 ID、所在考试 ID、用户 ID、得分、客观题得分、是否已判卷、是否已机器判卷。成绩单的实体图如图 1.9 所示。

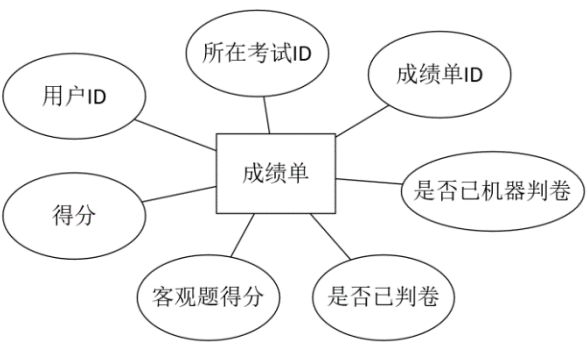


图 1.9 成绩单实体图

1.2.2 数据库逻辑设计

如前文所述，系统中的主要实体有：用户、课程、讨论、回复、考试、答卷、成绩单等。

(1) 用户数据表

表 1.2 用户数据表

字段名	数据类型	是否为空	备注
user_id	varchar(24)	NO	用户 id
password	varchar(70)	NO	密码
username	varchar(255)	YES	用户名
telephone	varchar(24)	YES	电话
email	varchar(24)	YES	邮箱
question1	varchar(255)	YES	密保问题 1
answer1	varchar(255)	YES	答案 1

续表 1.2 用户数据表

字段名	数据类型	是否为空	备注
-----	------	------	----

question2	varchar(255)	YES	密保问题 2
answer2	varchar(255)	YES	答案 2
user_avatar	varchar(255)	YES	用户头像 URL

(2) 课程数据表

表 1.3 课程数据表

字段名	数据类型	是否为空	备注
course_id	int	NO	课程 id
course_name	varchar(70)	NO	课程名称
user_id	varchar(24)	NO	创建者 id
is_start	bit(1)	YES	是否激活
is_public	bit(1)	YES	是否公开
cover_img	varchar(255)	YES	课程封面 URL
describes	varchar(255)	YES	课程简介
create_time	datetime	YES	创建时间

(3) 讨论数据表

表 1.4 讨论数据表

字段名	数据类型	是否为空	备注
discussion_id	int	NO	讨论 id
course_id	int	NO	所在课程 id
user_id	varchar(24)	NO	发布用户 id
title	varchar(60)	YES	标题
content	varchar(2047)	YES	内容
hits	int	YES	点击量
likes	int	YES	点赞数
create_time	datetime	YES	创建时间

(4) 回复数据表

该表中保存了用户的回复。回复具有发表用户，不一定有“@”用户。回复可以是对讨论进行回复，也可以是对章节进行回复，只能二者选其一，不能同时为空。

表 1.5 回复数据表

字段名	数据类型	是否为空	备注
reply_id	int	NO	回复 id
discussion_id	int	YES	对应讨论 id
chapter_id	int	YES	章节 id
from_uid	varchar(24)	NO	发表回复用户 id
to_uid	varchar(24)	YES	被回复的用户 id
content	varchar(2000)	YES	内容
create_time	datetime	YES	创建时间

(5) 考试数据表

表 1.6 考试数据表

字段名	数据类型	是否为空	备注
exam_id	int	NO	考试 id
course_id	int	YES	所在课程 id
author_id	varchar(24)	YES	发布者 id
type	int	YES	类型
title	varchar(255)	YES	标题
detail	varchar(2047)	YES	详情
not_order	int	YES	乱序出题
begin_date	datetime	YES	开始日期
end_date	datetime	YES	结束日期
create_time	datetime	YES	发布时间

(6) 答卷数据表

答卷表用于存放考生提交的答案，表中每条记录与考试题目一一对应，通过用户 id 和考试 id 可以筛选出用户对某个考试的全部答案数据，这样就组成了一份完整的答卷。答卷中通过题目类型和题目 id 来定位题目，用于判卷时查找题目的标准答案。

表 1.7 答卷数据表

字段名	数据类型	是否为空	备注
answer_id	int	NO	答卷 id
exam_id	int	YES	对应考试 id
subject_type	int	YES	题目类型
subject_id	int	YES	题目 id
subject_num	int	YES	题目序号
user_id	varchar(24)	YES	答题用户 id
answer_num	int	YES	答案数量
answer	varchar(2047)	YES	答案内容
score	int	YES	得分

(7) 成绩单数据表

成绩单数据表中保存了用户在某场考试中的得分数据。通过考试 id 和用户 id 可以唯一确定一条成绩单数据。当用户提交答卷时，系统会自动给用户创建一条成绩单数据，并进行机器判卷，计算出客观题得分保存到数据库中。这时成绩单中的机器阅卷选项值为 1，但主观题还需要老师评判，所以判卷项为 0，总分也为 0。当老师完成主观题评分并提交阅卷后，用户的总分会被保存到数据库中，并将已判卷项置为 1，含义为 true。

表 1.8 成绩单数据表

字段名	数据类型	是否为空	备注
score_id	int	NO	成绩单 id

exam_id	int	YES	考试 id
user_id	varchar(24)	YES	用户 id
score	int	YES	总分
objective_score	int	YES	客观题得分
is_checked	bit(1)	YES	是否完成判卷
system_checked	bit(1)	YES	是否完成机器判卷

1.3 登录相关模块设计

1.3.1 登录模块

登录模块是系统的入口模块，系统中大部分的数据操作和功能在使用时都要求用户必须处于登录状态，否则拒绝提供服务并跳转到登录页面。

在登录页面，用户填写自己的用户名、密码和验证码。首先对表单进行初步验证，将输入格式有误的数据排除并要求用户重新输入。确认无误后通过 `push` 请求将表单发送至后端验证。后端接收到登录请求，对比用户 `session` 中验证码的答案，正确则执行用户名与密码校验，否则返回验证码错误。登录页面设计如图 1.10 所示。

1.3.2 注册模块

新用户初次登陆时需要注册账号，前端对输入数据校验无误后请求注册服务。注册时先对用户名进行唯一性判定，用户名已存在则返回注册失败。否则保存新用户数据到数据库。

用户注册成功时，系统会自动给新用户添加计算机组成原理及相关课程，便于用户登录后学习。同时也会给用户添加新手教程，在用户登录可以查看教程来快速适应系统操作。新用户注册流程如图 1.11 所示。

1.3.3 找回密码模块

该模块中，用户可以事先设置好密保问题，当忘记密码时通过密保问题验证身份，能够重置密码。系统首先接收用户欲找回的用户名并查询该账户信息，成功则进入密保验证界面，否则返回用户名不存在。用户在密码界面输入密保答案，系统对答案进行匹配，成功则启动重置密码服务，否则返回验证失败。



图 1.10 登录页面设计

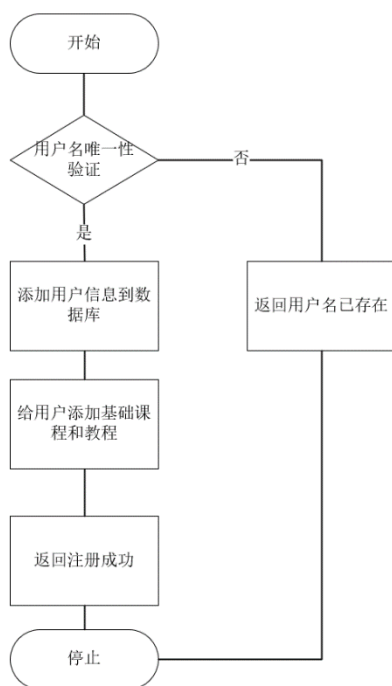


图 1.11 用户注册流程

1.4 用户首页相关模块设计

1.4.1 用户首页

用户首页中展示用户已加入的课程及相关通知,该页面也是教师端和学生端系统的入口。页面中可以切换显示用户加入或创建的课程。页面左上角的文字可以直达平台首页,右端菜单可以修改用户信息。页面顶部展示用户加入课程最近的 8 条通知,每条通知可以展开查看,支持浏览量监测。在查询用户加入的课程时,系统从查询结果中屏蔽私有和关闭的课程,其查询流程如图 1.12 所示。用户首页的设计如图 1.13 所示。

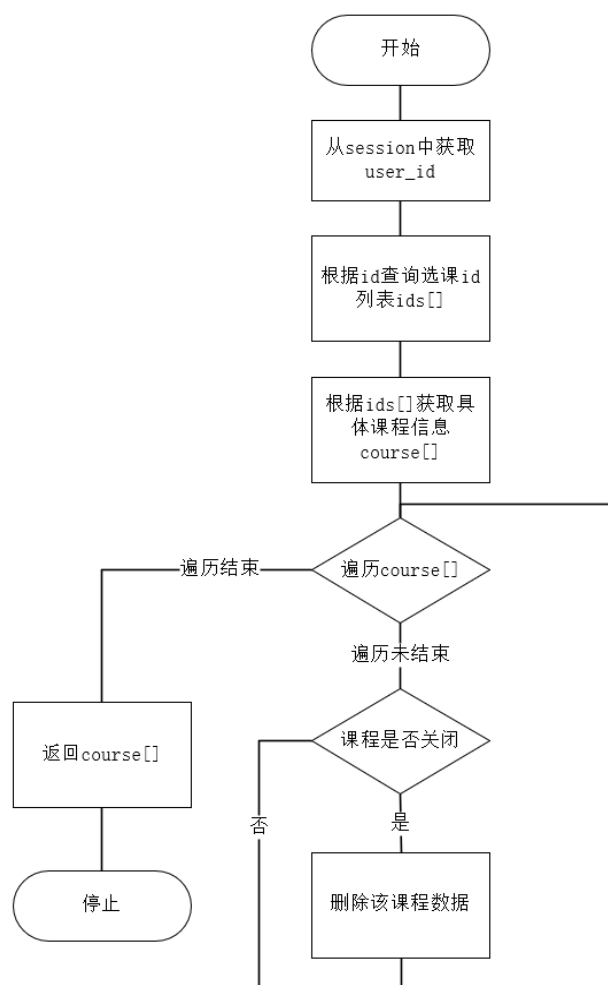


图 1.12 查询用户选课流程



图 1.13 用户首页

1.4.2 加入课程

通过该模块，用户可以加入课程，但无法加入未激活或者设置为私有的课程。当用户进入选课窗口时需要展示出用户未选课程以供用户选择。这里采用对课程

表的全查询，嵌套上一模块的功能，从全部课程中排除用户已选课程后即为未选课程。加入课程页面设计如图 1.14 所示。

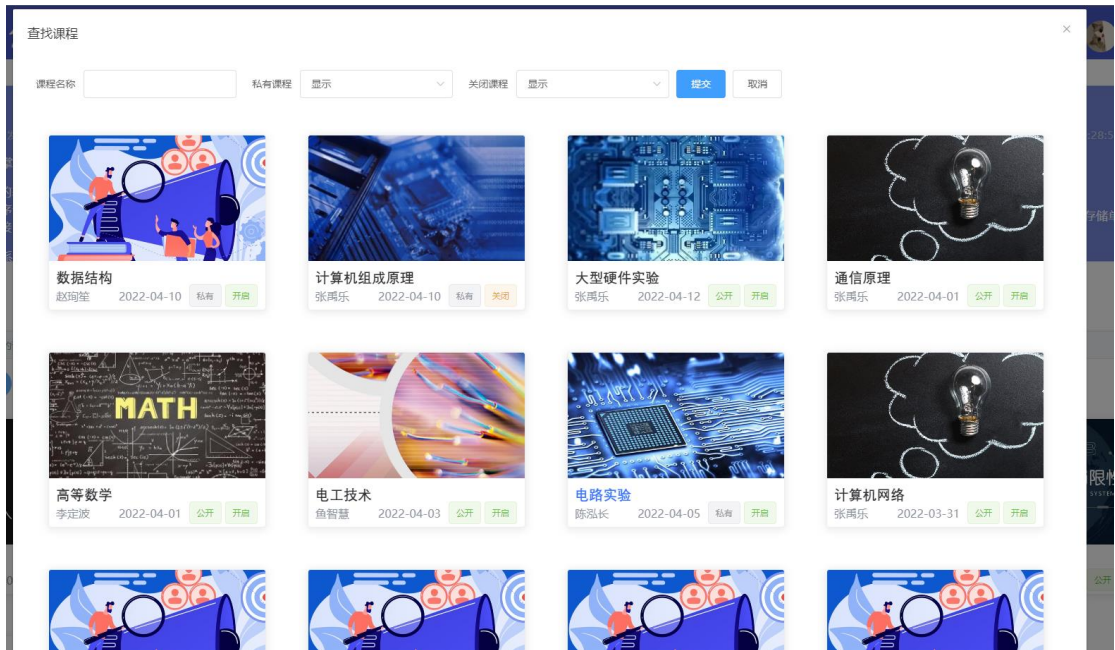


图 1.14 查找课程并加入

1.4.3 多条件查询课程

用户通过课程名查询课程，系统收到用户输入的课程名称后为其封装对应的 QueryWrapper，在课程表中进行模糊查询并返回查询结果集，前端再依据 isPublic 和 isStart 选项对该结果集进行过滤，并展示筛选后的课程数据。

对于查询所用实体类中的布尔类型，数据库中统一使用 bit 类型存放 1 和 0 来表示 bool 值。反应到后端代码中时，都为其封装了 bool 方法。示例代码如下：

```
private Integer isStart; // 是否开启
private Integer isPublic; // 是否公开
public Boolean getIsPublicBool(){
    return isPublic == 1;
}
public Boolean getIsStartBool(){
    return isStart == 1;
}
```

1.4.4 创建课程

用户可以创建课程，为课程设置名称和简介，上传封面或者使用系统提供的

默认封面。前端接收到用户输入的课程名称、简介、封面等数据后使用 `post` 请求发送到后端，后端对数据内容进行校验，封装到 `Course` 对象中。包括为课程分配封面数据，当用户未设置封面时还需要给课程初始化默认封面，同时将服务器时间作为课程创建时间封装到 `Course` 对象中。最后通过 `CourseService` 中的 `save` 方法将 `Course` 对象保存到数据库中，并返回创建成功消息。创建课程页面设计如图 1.15 所示。

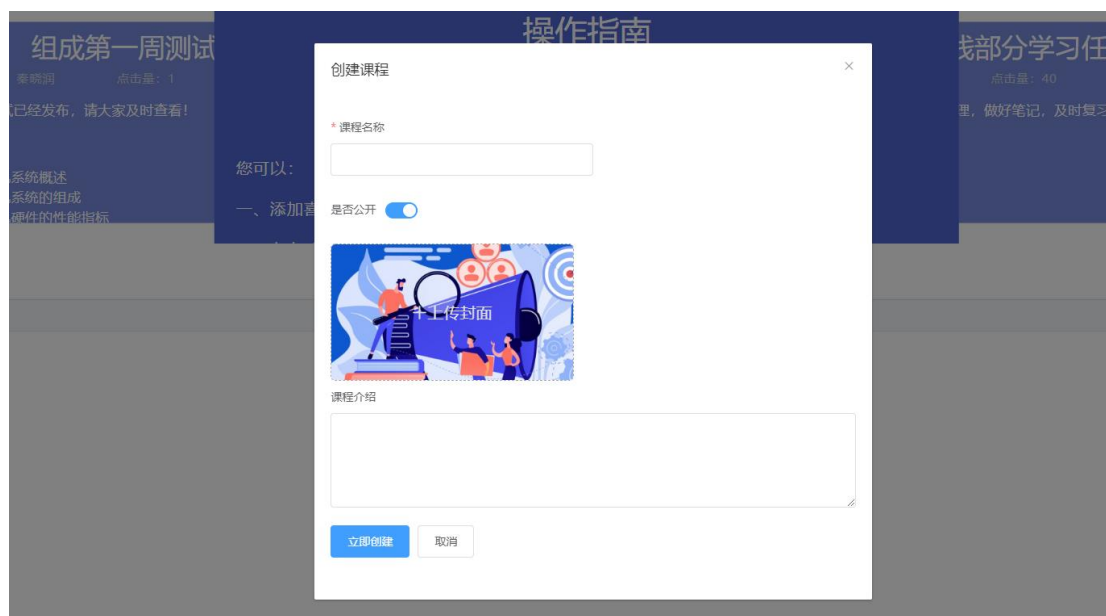


图 1.15 创建课程

1.4.5 查询课程通知

用户界面中需要显示用户所加入课程的通知中最新的 8 条通知，故首先需要在用户课程表中根据 `user_id` 获取其所在课程的 `ids` 列表，并根据该列表条件查询通知信息表。查询通知时用 `order by` 子句按照 `create_time` 字段倒序排序，通过 `limit` 关键字取前 8 条数据并返回 `List` 列表。

在该部分的实现中，SQL 操作比较复杂，单纯的组合 MP 接口会增加代码逻辑编写工作量，故这里使用 XML 配置方式直接编写 SQL，可以大大减轻业务逻辑层的工作量，最后的成品 SQL 语句如下：

```
select * from th_notice_view where course_id in
(select course_id from th_course_user where user_id = #{userId})
order by create_time desc
limit 0,#{size}
```

1.5 教师相关模块设计

1.5.1 课程管理模块

该模块中教师可以修改课程的加入条件和课程简介。前端通过 Dialog 对话框接收用户输入，由于课程状态 `isPublic` 在前端以 `bool` 类型被使用，所以在持久化时需要单独进行类型转换，后续其他类似变量的操作方式和转换方式均与此处相同。课程信息修改流程如图 1.16 所示。

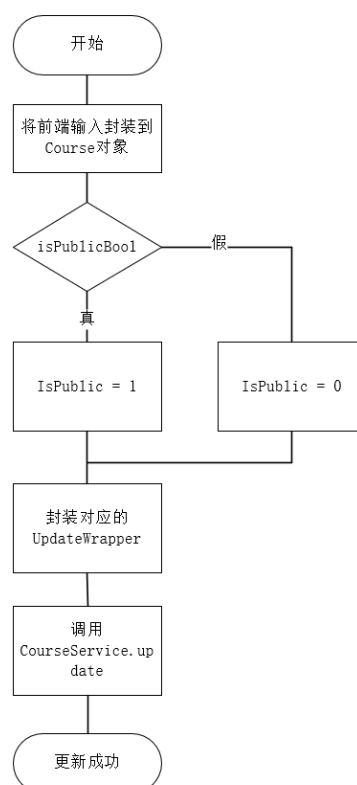


图 1.16 课程信息修改流程

1.5.2 课件管理模块

该模块中，用户上传的文件以字节流形式发送到后端，后端通过 Spring 提供的 `MultipartFile` 工具类进行接收。这里系统需要给用户上传的每个文件分配一个唯一标识，并将该标识拼接到文件名中从而使得用户可以通过文件名直接索引到该文件，这里使用文件上传到系统的时间戳后 5 位作为文件的唯一标识。

系统会给每个用户创建一个目录来存放该用户相关的所有文件，在用户上传文件时，文件将被存储到用户目录下的 `resource` 目录中，如果目录不存在还需要自动创建该目录。目录创建后的效果如图 1.17 所示。图中“64645_任务书.docx”中的前 5 位数字即为文件的唯一标识符，该标识符和文件名一起组成文件的唯一

索引。

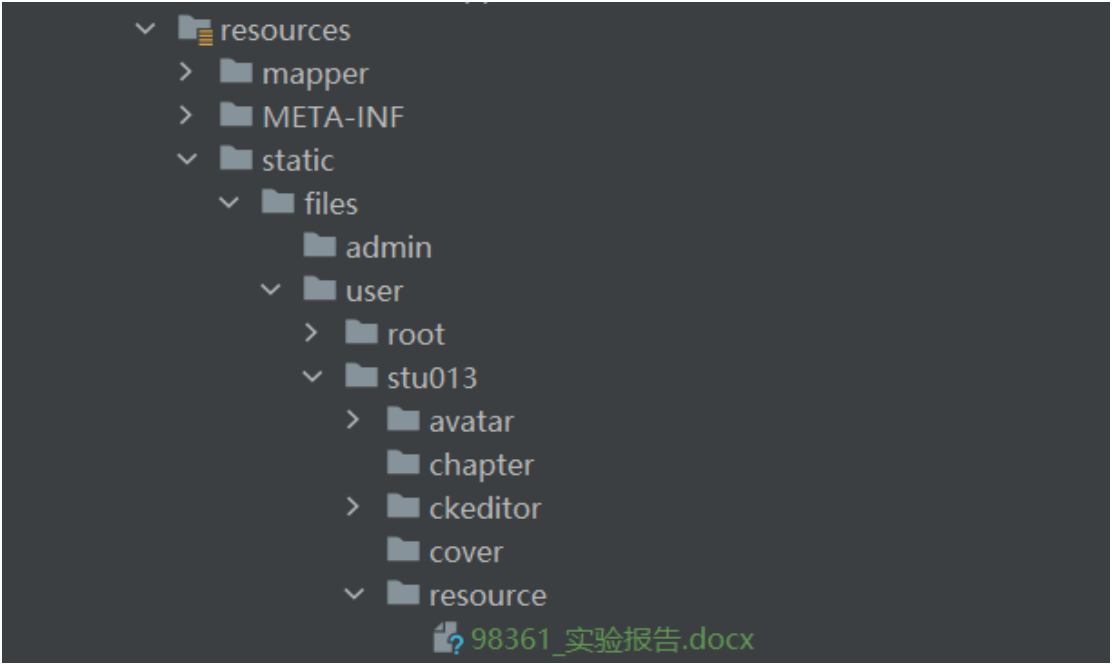


图 1.17 用户资源目录结构

在教师下载文件时，系统根据文件的唯一索引到该课程创建者的目录中查询该文件，并将文件转为字节流发送到前端，前端再使用<a>标签的下载功能将文件下载到本地。课件管理页面设计如图 1.18 所示，其中文件名前的标识串可隐藏。

← 返回 | 资料

添加资料 批量下载 批量删除

搜索内容 文件名 查询 重置

<input type="checkbox"/>	序号	文件名	文件大小	下载量	创建日期	操作
<input type="checkbox"/>	1	28715_第4讲-1+存储器1.ppt	4.97MB	0	2022-05-20 18:08:50	下载 删除
<input type="checkbox"/>	2	28915_第07讲-1+控制单元的功能.ppt	1.45MB	0	2022-05-20 18:08:50	下载 删除
<input type="checkbox"/>	3	28935_第06讲+CPU+的结构和功能.ppt	4.71MB	0	2022-05-20 18:08:50	下载 删除
<input type="checkbox"/>	4	28965_第07讲-2+控制单元的设计.ppt	2.22MB	0	2022-05-20 18:08:50	下载 删除
<input type="checkbox"/>	5	28095_第02讲+计算机的发展及应用.ppt	1.04MB	0	2022-05-20 18:08:49	下载 删除
<input type="checkbox"/>	6	28442_第4讲-4+辅助存储器.ppt	1011.5KB	0	2022-05-20 18:08:49	下载 删除
<input type="checkbox"/>	7	28452_第4讲-3+存储器3.ppt	1.29MB	0	2022-05-20 18:08:49	下载 删除
<input type="checkbox"/>	8	28656_第3讲 数值及运算.pptx	1.92MB	0	2022-05-20 18:08:49	下载 删除
<input type="checkbox"/>	9	28656_第05讲+指令系统.ppt	2.08MB	0	2022-05-20 18:08:49	下载 删除

图 1.18 课件管理

1.5.3 章节管理模块

该模块的设计基于前文中的文件上传功能。教师将章节上传至服务器端后，服务器端将章节信息封装为 Chapter 对象并从 session 中获取 userId，然后调用持久层接口 ChapterService 的 save 方法将章节信息保存到数据库中。该模块页面设

计如图 1.19 所示。

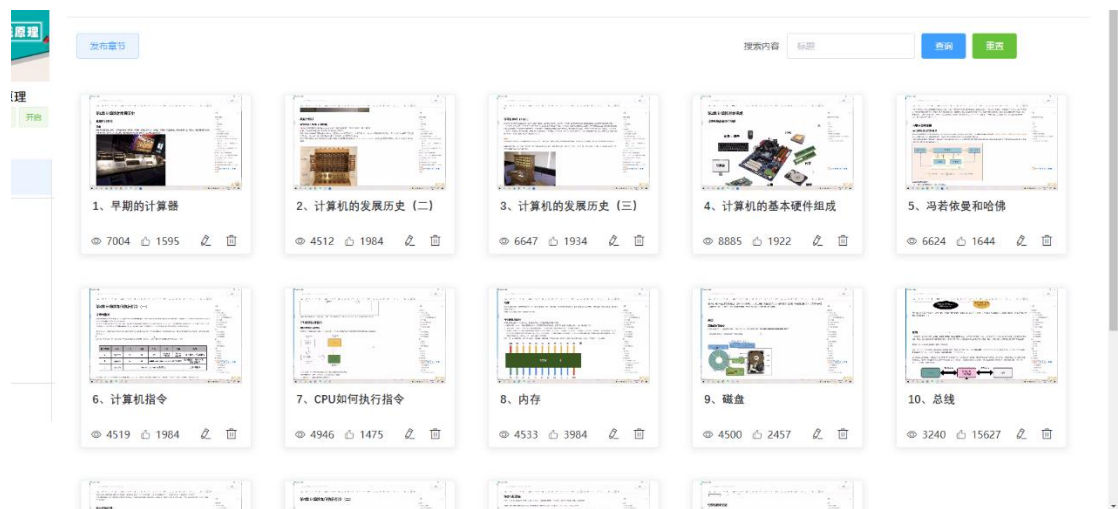


图 1.19 章节视频管理

用户浏览章节内容时，系统在用户进入页面 5 秒后执行增加浏览量的方法，实现对章节浏览量的自增功能。该方法在教师模块中不生效。

章节中还添加了评论区功能，教师输入评论后前端将评论数据发送到后端。后端将评论数据封装为 Reply 对象并向其中添加 toUid，从 session 中获取 userId 作为 fromUid。以服务器时间为评论创建时间，调用 ReplyService.save 将 Reply 对象持久化，最后能够在视频底部展示章节所含评论。

1.5.4 讨论管理模块

教师可以在这里发布讨论，管理课程内所有讨论，以及查看讨论内容，在评论区和学生互动。创建讨论时，教师在编辑窗口中输入讨论标题和内容，前端使用 post 方式发送请求到后端。后端将前端发送的讨论数据封装为 Discussion 对象，从 session 中获取 userId，并初始化该对象中的 hits 和 likes。再调用 DiscussionService.save 方法将数据保存到数据库。

教师和学生讨论模块中发表评论并进行点赞，此处评论区实现方法同前文中的章节评论区一致。点赞功能的实现基于 th_chapter_like_records 表。用户给讨论点赞后会添加一条点赞数据到 th_chapter_like_records 表中，故当用户给内容点赞时，系统先去查询该表，根据结果数判断用户是否已经赞过该内容，如果结果数大于 0 则无法进行点赞，前端将点赞按钮高亮，此时点赞按钮的功能更改为取消点赞；若结果数为 0 则保存一条点赞数据到 th_chapter_like_records 表中，表示用户已赞过该内容。

模块实现如图 1.20、1.21 所示。

新建讨论						
搜索内容 标题 筛选 重置						
序号	标题	发布者	点击量	点赞数	创建日期	操作
1	这是一条学生讨论	林诗倩	14	1	2022-05-03 08:40:10	编辑 删除
2	作业会谈	陈时文	28	1	2022-05-02 16:03:12	编辑 删除
3	队列的理解	陈时文	35	19	2022-05-01 17:14:28	编辑 删除
4	递归的理解	陈时文	64	25	2022-05-01 17:13:51	编辑 删除

图 1.20 讨论管理

繁星之夜

林诗倩

返回 讨论

测试课程

2022-04-10 公开 开始

章节

通知

课件

讨论

实验

考试

设置

发布者：林诗倩 2023-07-11 10:31:39

栈和队列

在线性表中，栈和队列是我们需要掌握的重要知识点。关于这两者的区别，大家可以在下面说说自己的看法吗

进栈

出栈

栈顶

栈底

8 2

点赞 评论

评论区

意北海 2023-07-11 15:00:41

栈 (Stack) 是限定仅在表尾进行插入和删除操作的线性表。
我们把允许插入和删除的一端称为栈顶 (top)，另一端称为栈底 (bottom)，不含任何元素的栈称为空栈。栈又称为后进先出 (Last In First Out) 的线性表，简称 LIFO 结构。
理解栈的定义需要注意：
首先它是一个线性表，也就是说，栈元素具有线性关系，即前驱后继关系。只不过它是一种特殊的线性表而已。定义中说是在线性表的表尾进行插入和删除操作，这里表尾是指栈顶，而不是栈底。
它的特殊之处就在于限制了这个线性表的插入和删除位置，它始终只在栈顶进行。这也就使得：栈底是固定的，最先进栈的只能在栈底。

评论

林诗倩 2023-07-11 10:32:52

栈与队列的相同点：
1. 都是线性结构。
2. 插入操作都是限定在表尾进行。
3. 都可以用顺序结构和链式结构实现。
4. 插入与删除的时间复杂度都是 $O(1)$ ，在空间复杂度上两者也一样。
5. 多链栈和多链队列的管理模式可以相同。
栈与队列的不同点：
1. 删除元素的位置不同，栈的删除操作在表尾进行，队列的删除操作在表头进行。
2. 应用场景不同：常见栈的应用场景包括括弧问题的求解、表达式的转换和求值、函数调用和递归实现、深度优先搜索遍历等；常见的队列的应用场景包括计算机系统中各种资源的管理、消息缓冲器的管理和广度优先搜索遍历等。
3. 顺序栈能够实现多栈空间共享，而顺序队列不能。

评论

Copyright © 2022 繁星工作室 Fan-Xin-Chen-izone

图 1.21 评论区管理

1.5.5 实验管理模块

用户可以在这里发布实验，上传实验报告，查看并下载学生的实验报告。该模块的实现基于 `th_experiment` 表和 `th_report` 表，教师编辑好实验标题后上传任务书到系统中，前端将实验信息通过 `post` 方式发送到后端，后端从 `session` 中获取 `courseId` 同前端的信息一起封装为 `Experiment` 对象，调用 `ExperimentService` 的 `save` 方法将数据保存到数据库。模块实现如图 1.22 所示。

前端展示的每一条实验都提供了查看学生报告的功能。该功能首先向数据库请求实验相关的所有报告，并以表格形式展示给教师。这里提供下载学生报告的功能同上文中文件下载功能的实现思路基本一致。模块实现如图 1.23 所示。



← 返回 实验		搜索内容 标题		查询	重置
新建实验					
序号	标题	任务书	创建日期	操作	
1	实验一、系统设计实验	± 64648_任务书.docx	2022-05-16 10:37:51	编辑	删除
2	实验二、存储器实验	± 8723_计算机组成实验报告 (1).doc	2022-05-20 18:30:34	编辑	删除
3	实验三、内存设计实验	± 90648_任务书 (1).docx	2022-05-20 18:31:44	编辑	删除
4	实验四、缓存设计实验	± 19590_任务书 (2).docx	2022-05-20 18:32:13	编辑	删除
5	实验五、地址管理实验	± 37844_任务书 (3).docx	2022-05-20 18:32:31	编辑	删除
6	实验六、寄存器	± 18521_任务书_寄存器.txt	2022-06-01 10:15:20	编辑	删除
7	实验七、存储器	± 34230_任务书_存储器.txt	2022-06-01 10:15:35	编辑	删除
8	实验八、ALU	± 45637_任务书-ALU.txt	2022-06-01 10:15:47	编辑	删除
9	实验九、ACC	± 82037_任务书-ACC.txt	2022-06-01 10:16:23	编辑	删除

图 1.22 实验管理



← 返回 实验		搜索内容 标题		查询	重置
新建实验					
序号	姓名	学生报告	创建日期		
1	陈长宏	± 23157_实验报告3	2022-06-16 18:20:18		
2	张玉端	± 54871_实验报告2	2022-06-16 18:19:51		
3	鱼皮	± 45782_实验报告4	2022-06-10 18:20:53		
4	张天禹	± 21865_实验报告5	2022-06-07 18:21:35		
5	何天祥	± 14568_实验报告1	2022-06-01 18:19:31		
6	1807004022	± 8013_测试数据.docx	2022-06-01 02:09:56		
7	李传国	± 55310_计算机组成实验报告 (1).doc	2022-05-16 11:34:06		
8	林诗倩	± 50990_测试数据.docx	2022-05-16 11:17:02		
9	陈时文	± 47844_测试数据.docx	2022-05-16 10:51:32		

图 1.23 学生报告下载

1.5.6 考试管理模块

用户能够在该模块中发布考试，设置考试标题、详情、起止时间、题目是否乱序展示等。可以对课程中的考试进行编辑和删除操作。可以给考试出题，考试开启后无法直接编辑考试信息。列表中的考试项可以排序和查找，支持分页展示。如图 1.24 所示。

返回

考试

发布考试

搜索内容

标题

查询

重置

序号	名称	发布者	打乱顺序	开始日期	截止日期	创建日期	操作
1	第一周测试	本人	<input type="checkbox"/>	2022-05-12 00:00:00	2022-05-14 00:00:00	2022-05-15 20:51:55	编辑 出题 删除
2	第二周测试	本人	<input checked="" type="checkbox"/>	2022-05-18 09:00:00	2022-05-19 11:00:00	2022-05-10 20:02:26	编辑 出题 删除
3	第三周测试	本人	<input type="checkbox"/>	2022-05-24 14:00:00	2022-05-25 14:00:00	2022-05-11 09:21:44	编辑 出题 删除
4	第四周测试	本人	<input checked="" type="checkbox"/>	2022-05-25 00:00:00	2022-05-26 00:00:00	2022-05-12 20:21:46	编辑 出题 删除
5	期中考试	管理员	<input checked="" type="checkbox"/>	2022-05-27 09:00:00	2022-05-27 12:00:00	2022-05-11 09:30:09	编辑 出题 删除
6	第五周测试	本人	<input checked="" type="checkbox"/>	2022-05-29 00:00:00	2022-05-30 00:00:00	2022-05-12 20:22:06	编辑 出题 删除
7	第六周测试	本人	<input type="checkbox"/>	2022-05-30 00:00:00	2022-05-31 00:00:00	2022-05-12 20:22:20	编辑 出题 删除
8	第七周测试	本人	<input checked="" type="checkbox"/>	2022-06-01 00:00:00	2022-06-06 00:00:00	2022-05-12 20:22:36	编辑 出题 删除
9	第八周测试	本人	<input type="checkbox"/>	2022-06-06 00:00:00	2022-06-10 00:00:00	2022-05-12 20:22:51	编辑 出题 删除

图 1.24 考试管理

用户可以在出题面板中给考试设计题目，题目类型包括选择、填空、判断、简答四种题型，每道题目可以单独设置分值和解析。考试未开始前，所有的题目都可以自由编辑，但考试开始后将只能查看考题，无法修改或者删除试题。如图 1.25 所示。考试结束后老师的判题模块如图 1.26、1.27、1.28 所示。

添加题目

选择题

填空题

判断题

简答题

单选

4分

题目

计算机系统中的存储器系统是指

- 1、ROM存储器
- 2、RAM存储器
- 3、主存储器
- 4、主存储器和外存储器

答案

解析

无解析

图 1.25 出题面板（部分）

1		陈时文	已交卷	已判卷	客观题: 41分	总分: 75分
2		陈长宏	已交卷	待评分	客观题: 32分	
3		张天禹	已交卷	已判卷	客观题: 41分	总分: 73分
4		林诗情	已交卷	已判卷	客观题: 63分	总分: 99分
5		鱼皮	未交卷			
6		李传国	未交卷			

图 1.26 判题区总览



图 1.27 单人判题面板展示



图 1.28 主观题判题模块

该教师创建考试时，前端携带考试相关数据请求后端的“/Exam/createExam”服务，该服务将前端发送的数据封装为 Exam 对象并调用 ExamService.save 方法将考试信息保存到数据库。教师可以编辑考试的题目信息，不同题目的出题 Dialog 包含不同的功能组件，但最后封装的数据格式是一致的。教师完成出题后，前端发送 Subject 题目列表到后端，后端循环遍历 Subject 列表，为每道题目添加 th_paper 数据，出题功能具体流程如图 1.29 所示。

考试中教师判题功能的实现基于 th_answer 表和 th_score 表。教师在判题模块中能够给学生的主观题进行评分。评分完成后前端将学生的得分数据发送给后

端，后端通过 examId 和 userId 从 th_score 表中确定学生得分记录并更新其 score 字段和 is_checked 字段。

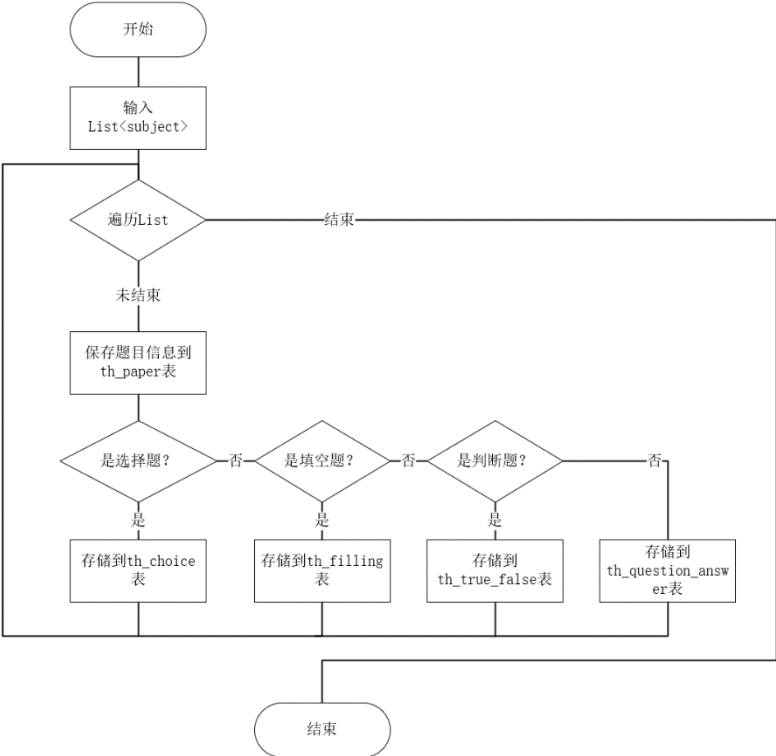


图 1.29 试卷持久化流程图

1.5.7 学生管理模块

该模块显示课程内所有学生的学习记录和相关信息，这些信息全部存储在用户和课程的中间表 th_course_user 里。教师可以在该模块查看、搜索、导出学生的学习数据。模块启动时，服务器从上表中将与本课程相关的学生数据全部查询出来，查询条件是记录的 course_id 与当前 session 中的 courseId 相等，查询结果返回 CourseUser 对象的列表。数据导出模块使用封装后的 TableExport 模块，接收参数为当前模块中的 el-table 对象。

1.6 学生相关模块设计

1.6.1 章节浏览模块

该模块中学生可以查看章节视频并点赞和评论。视频链接从 th_chapter 表中读取，前端使用<video>标签显示视频，添加 control 选项后还可以提供倍速播放和画中画播放。用户给章节点赞和评论的实现思路同教师端的类似功能基本一致。章节的显示使用 Element UI 中的<el-card>标签实现。该模块设计如图 1.30 所示。

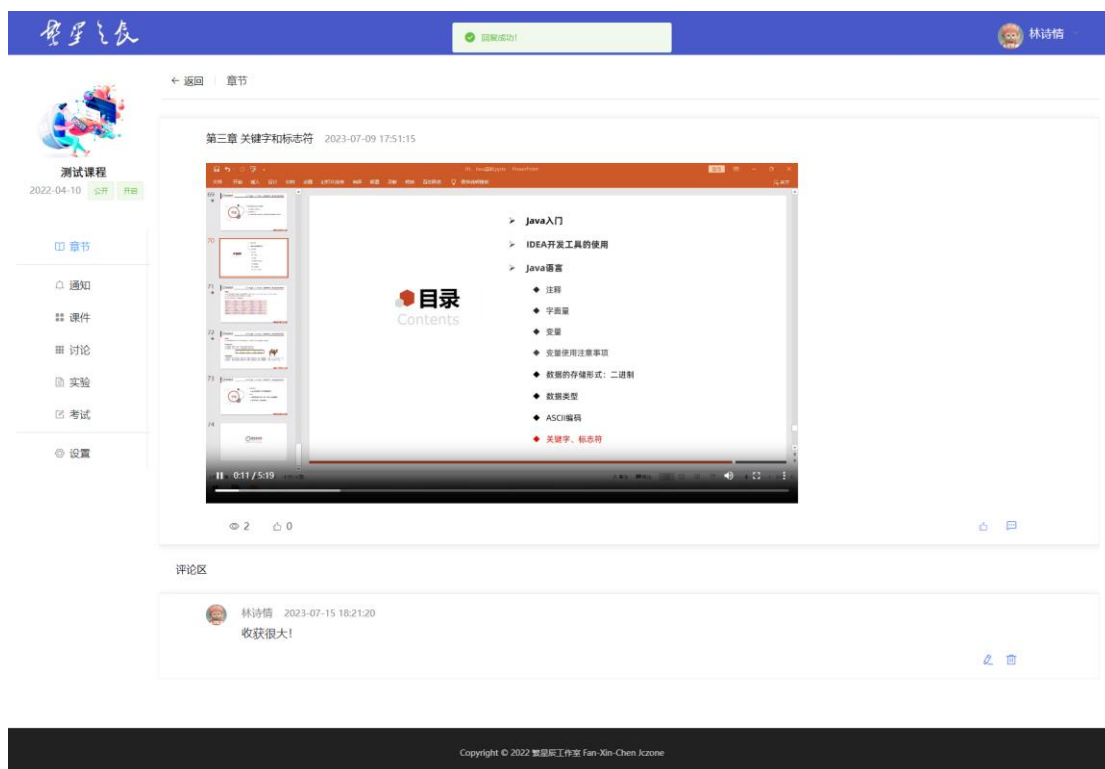


图 1.30 学生章节模块

1.6.2 课件下载和通知模块

学生可以下载教师上传的课程，每次下载，文件的下载量一同增加。学生每次浏览课程通知时，通知的浏览量也要增加。两者的实现都是在学生点击相关按钮时进行，并操作对应的数据库。

1.6.3 讨论模块

学生在该模块中可以参与/发布讨论。对自己创建的讨论具有编辑功能，由于创建和编辑讨论的编辑窗口是相同的，所以在实现时两者可以共用一个 Dialog 组件，但不同按钮激活 Dialog 时，组件的初始化和提交的反应事件不同。这里使用一个 bool 变量 isEditState 来对创建和编辑状态做区分。这种设计思想在其他模块中也很常见，基本上同时具备对某项数据创建和编辑功能的模块，该功能都采用这种思想，比如教师端的考试创建模块，通知管理模块等。该模块设计如图 1.31 所示。



图 1.31 学生讨论

1.6.4 实验模块

在教师发布实验以后，课程内的学生都可以对该实验提交报告，但只能提交一份报告。所以在设计学生端时，不同于教师的两层页面结构，学生端只需要一层页面即可。即直接在实验列表中对实验上传报告，而不是点进实验内部进行上传。这样就需要同时展示实验的信息和学生上传的报告信息，后端就需要通过字段 `experimentId` 来对 `Experiment` 对象和 `Report` 对象进行连接，得到其 `ExperimentPlus` 对象，再展示到前端界面。三种类的属性如表 1.9 所示。

表 1.9 属性对照表

Experiment	Report	ExperimentPlus
<code>experimentId</code>	<code>reportId</code>	<code>experimentId</code>
<code>courseId</code>	<code>experimentId</code>	<code>reportId</code>
<code>title</code>	<code>userId</code>	<code>courseId</code>
<code>fileName</code>	<code>fileName</code>	<code>userId</code>
<code>createTime</code>	<code>createTime</code>	<code>title</code>
		<code>fileName</code>
		<code>reportFile</code>
		<code>createTime</code>

1.6.5 考试模块

不同于其他模块中对数据的传输是通过封装数据库中的表为实体类，直接读取数据库内容到相应的对象中来传输。考试模块中的题目需要特别封装为 `Subject` 列表进行传输，学生的答案则封装为 `Solution` 列表进行传输。这是由于试题中存在四种题型，不同题型在数据库里所含字段并不完全相同，如果对每种题型都封装对应的答卷表，操作起来比较繁琐，传输题型也很麻烦。所以我归纳了

每种题型的共同点，创建了通用的数据传输模板 **Subject** 和 **Solution** 用来进行规范化的数据传输。他们的属性如表 1.10、表 1.11 和表 1.12 所示。

表 1.10 选择题和填空题属性表

Choice	备注	Filling	备注
subjectId	题目 id	subjectId	题目 id
content	内容	content	内容
answerNum	答案数	answerNum	答案数
analysis	解析	analysis	解析
score	分值	score	分值
		isOrder	是否有序
		isCapital	是否大写

表 1.11 判断题和简答题属性表

TrueFalse	备注	QuestionAnswer	备注
subjectId	题目 id	subjectId	题目 id
content	内容	content	内容
answer	答案数	answer	答案数
analysis	解析	analysis	解析
score	分值	score	分值

表 1.12 封装类属性表

Subject	备注	Solution	备注
paperId	试题 id	answerId	答案 id
subjectId	题目 id	examId	试题 id
subjectNum	题目序号	subjectId	题目 id
subjectType	题目类型	subjectType	题目类型
choice	选择题对象	subjectNum	题目序号
filling	填空题对象	userId	考生 id
questionAnswer	简答题对象	answerNum	选择/填空的项数
trueFalse	判断题对象	answerOptions	选择/填空的答案列表
choiceOptions	选择题选项列表	answer	用户答案
fillingBlanks	填空题空项列表	score	用户得分

学生输入答案并确认提交后，前端使用 **post** 请求将答案以 **Solution** 列表的形式发送到后端。后端对该列表进行读取，解析其中每项 **solution** 的题目类型和答案，根据题型进行客观题评分并记录到用户的 **score** 表中。对于选择题和填空题，还需要解析他们的选项/空项答案。具体处理流程如图 1.32 所示。学生答题模块设计如图 1.33、1.34 所示。

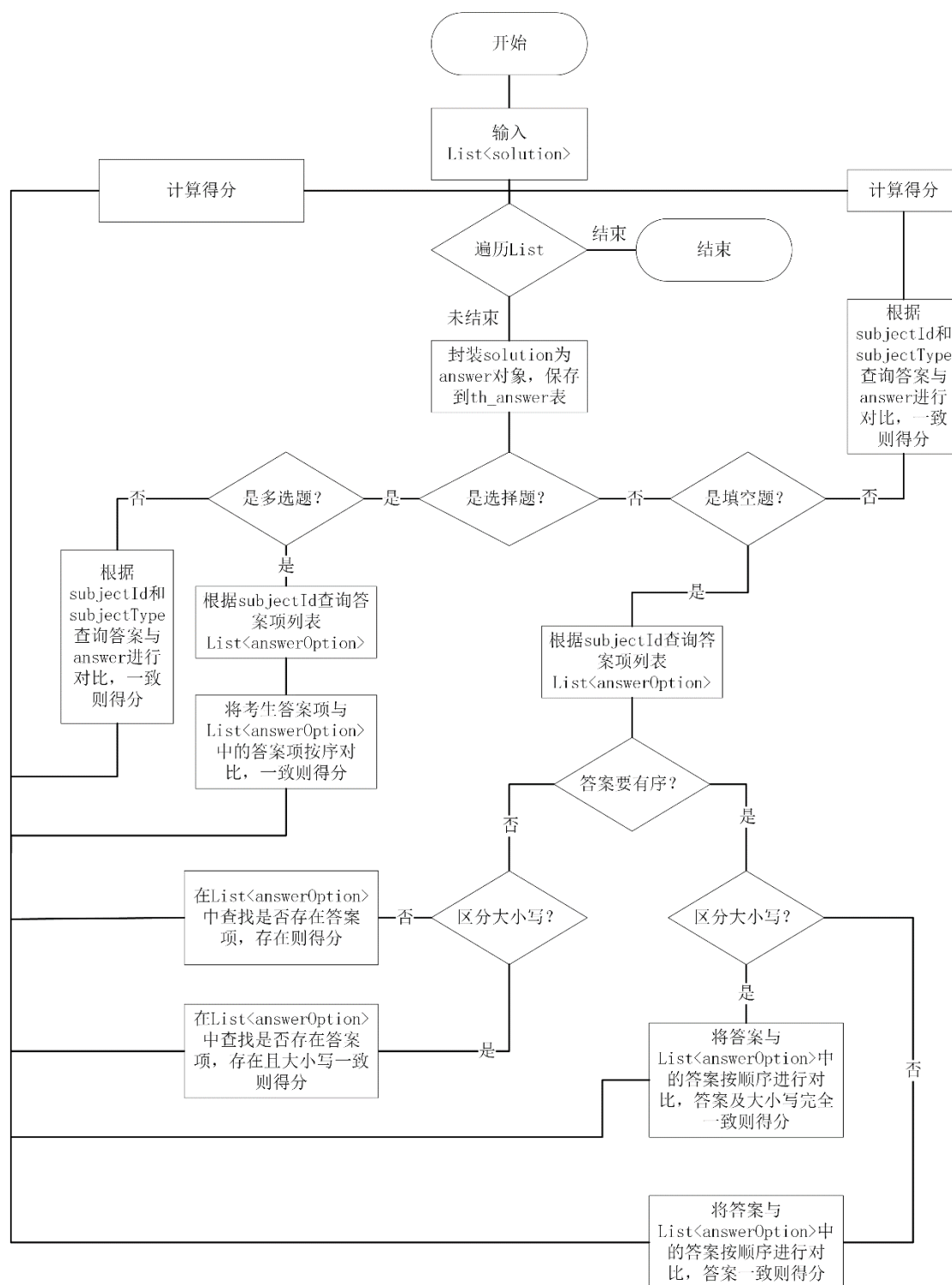


图 1.32 答卷持久化流程

15

填空

4分

题目

输入输出设备数据传输控制方式有五种，比如程序查询方式、____、____、____、____、

回答:

答案1

程序中断方式

答案2

直接存储器存取方式

答案3

通道控制方式

答案4

外围机处理方式

图 1.34 主观题答题界面 (部分)

1.7 管理员项目模块设计

1.7.1 课程管理模块

该模块提供对课程数据的全局管理权限。管理员可以对系统中所有课程及课程相关数据进行管理，模块中的功能基本是对教师管理功能的复用和修改。如图 1.35 所示。



序号	名称	创建者	状态	创建日期	操作
1	计算机应用技术	鱼智慧	开启	2022-04-18 22:26:26	管理 删除
2	大型硬件实验	张周乐	开启	2022-04-12 19:53:44	管理 删除
3	操作系统	李定波	关闭	2022-04-11 19:54:16	管理 删除
4	数据结构	赵瑞莲	开启	2022-04-10 20:57:39	管理 删除
5	测试课程	赵瑞莲	开启	2022-04-10 20:57:08	管理 删除
6	计算机组成原理	张周乐	关闭	2022-04-10 06:13:36	管理 删除
7	计算系统的局限性	秦晓润	开启	2022-04-10 06:12:10	管理 删除
8	单片机原理	秦晓润	开启	2022-04-07 00:39:08	管理 删除
9	微机原理与汇编	秦晓润	开启	2022-04-07 00:38:37	管理 删除
10	计算机组成原理	秦晓润	开启	2022-04-07 00:37:59	管理 删除

图 1.35 课程管理

1.7.2 用户管理模块

该模块可以完成用户数据的筛选和导出，导出功能依靠对 Excel-Export 组件的封装实现。该组件中使用了xlsx和file-saver两个依赖中的方法，可以对Element UI中的表格数据进行导出。如图 1.36 所示。



序号	账号	姓名	联系方式	邮箱	操作
1	tea004	陈冠长	10000000011	10000000011@163.com	修改密码 注销
2	tea003	鱼智慧	10000000010	10000000010@163.com	修改密码 注销
3	tea002	张周乐	10000000009	10000000009@163.com	修改密码 注销
4	tea001	李定波	10000000008	10000000008@163.com	修改密码 注销
5	stu043	张玉坤	10000000007	10000000007@163.com	修改密码 注销
6	stu033	何建祥	10000000006	10000000006@163.com	修改密码 注销
7	stu023	赵瑞莲	10000000005	10000000005@163.com	修改密码 注销
8	stu022	温慧月	10000000004	10000000004@163.com	修改密码 注销
9	stu016	stu016			修改密码 注销
10	stu013	林诗儒	10000000003	10000000003@163.com	修改密码 注销

图 1.36 用户管理