

毕 业 论 文

题 目 计算机组成原理培训学习平台

学 院 计算机科学与软件学院

专 业 计算机科学与技术

年 级 2019级

学 号 201924111303

学生姓名 卢津瑜

指导教师 王俊波

完成时间 2023 年 4 月

注：指导教师一栏：第一导师为校内指导老师，第二导师为企业或实习学校的指导老师；

（若题目过长，可以用两行；删除红色字体，空三行）

肇庆学院教务处制

**学术诚信声明**

本人所呈交的毕业论文，是在指导教师的指导下独立完成。研究工作所取得的成果、数据、图片资料均真实可靠。除文中已注明引用的内容外，不包含任何其他人或集体已经发表或撰写过的作品或成果。对本论文的研究做出重要贡献的个人和集体，均已在文中以明确的方式标明。本毕业论文的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

本人签名（手写）： 日期： 年 月 日

**目 录**（小二号黑体，居中排段前，段后30磅）

摘要与关键词………………………………………………………………………………1

[1 绪论(一级标题) 1](#_Toc192066266)

[1.1 课题背景(二级标题) 1](#_Toc192066267)

[1.1.1 三级标题 1](#_Toc192066268)

[1.2 技术现状 1](#_Toc192066269)

[6 系统开发环境 2](#_Toc192066270)

[6.1 本课题的硬件系统 2](#_Toc192066271)

[6.2.1\*\*\*\*\*\*\*\*\*\*\*\*\*\* 2](#_Toc192066272)

7 结论………………………………………………………………………………………2

参考文献……………………………………………………………………………………2

Abstract and Keywords…………………………………………………………………..…2

致 谢………………………………………………………………………………………2

**注：**

**1、排版自动生成目录，小四号宋体，行间距20磅，原则上不超过三级标题）**

**2、目录包含摘要与关键词、一二三级标题、结论、参考文献、英文摘要与关键词、附录（可节选）、致谢等。）**

**毕业论文装订顺序！ （**红色、蓝色字为格式说明，请删除本处说明**）**

封面→学术诚信声明→目录→论文正文（**中文题目→中文摘要及关键词→绪论或引言→正文→结论→参考文献→英文题目→英文摘要及关键词**）→致谢→附录（必要时）。（注意：中英题目下及正文中不要写指导老师和学生的姓名）

**封面（封底）、学术诚信声明、目录、致谢实行单面打印；**

**中文题目及摘要与关键词、绪论或引言、正文、结论、参考文献、英文题目及英文摘要与关键词、附录实行双面打印。**

计算机组成原理培训学习平台的设计实现（小二号，黑体居中排段前后各30磅）

摘 要**:** 计算机组成原理是计算机专业的核心课程，课程中许多的知识点抽象而且难以理解，并且大多都具有较高的技术性，课程难度较大。且在整个课程体系中占有重要地位。而计算机组成原理培训学习平台是能够提供给学生线上做好课前预习和演练，具有视频学习和习题演练功能，对于复杂的课程内容可以做到及时消化和吸收，也方便教师跟踪和反馈。本系统采用SpringBoot和JQuery技术，还有一下特点：（1）使用MongoDB数据库，能处理高并发的读写和访问请求。（2）对于视频等大文件的上传，采用了文件分片上传、大文件秒传、断点续传的技术，使得上传文件更加安全且快速。 (“摘要”两个字空两格，**黑体; 摘要内容：小四号宋体，行间距20磅，300字以内) （**“关键词”：**黑体，小四号；关键词之间用分号“; ”最后一词不加标点）**

关键词:计算机组成原理; SpringBoot; 分片上传;JQuery

（空行，小四，删除此注释）

**Abstract**: The principle of computer composition is the core course of computer major, many knowledge points in the course are abstract and difficult to understand, and most of them are of high technology, the course is more difficult. And plays an important role in the whole curriculum system. The computer composition principle training and learning platform can provide students with online pre-class preview and drill, with video learning and exercise drill functions, which can timely digest and absorb complex course content and facilitate teachers to track and feedback. This system adopts SpringBoot and JQuery technologies, and has the following features: (1) It uses MongoDB database, which can handle highly concurrent read and write and access requests. (2) For uploading large files such as videos, it adopts the technologies of file fragment uploading, second uploading of large files, and resumable uploading of breakpoint, making file uploading safer and faster.

**Keyword**: principles of computer composition; SpringBoot; JQuery; Fragment upload

**注释：中英文题目下不得出现指导老师或作者的姓名；**

**1、英文题目：小二号，单倍行距、段前后各30磅，Time New Roman加粗居中；**

**2、“Abstract、Keywords”两个词: 小四号 Time New Roman，加粗顶格；**

**3、英文摘要内容：（小四号，Times New Roman字体，行间距20磅，标点符号采用英文标点）；**

**4、英文关键词（3-5个）：与中文关键词对应；单个词组，首字母大写；**

# 1 绪论（一级标题，行间距20磅、段前、段后各24磅，数字后留1空格）

## 1.1 课题背景（二级标题小四号黑体，行间距20磅、段前段后各12磅，数字后留1空格）

现在的信息技术不断变化升级，越来越丰富的网站以及应用使得用户更多地愿意接受在线培训平台的学习。在线学习，在线培训一直都是“互联网+”大环境下广受关注的领域之一。计算机组成原理是计算机专业的核心课程，课程中许多的知识点抽象而且难以理解，并且大多都具有较高的技术性，课程难度较大，且在整个课程体系中占有重要地位。采用线下实时教学，学生无法进行较好的课前预习和演练，对于复杂的内容无法做到及时消化和吸收，而且学生对该课程的掌握程度也无法直观的看到，不利于教学的反馈和跟踪，且线下的教学在遇到突发情况时，无法保证正常的教学，严重影响教学进度。

（正文宋体，行间距20磅，正文中的英文字符一律用 Times New Roman字体 ，标题中的英文字符一律用Arial字体，参考文献用上标表示）

### 1.2 课题意义（小四号黑体，行间距20磅、段前、段后各6磅，数字后留1空格）

在此背景下，本文提出了计算机组成原理课程培训学习平台的设计与实现，本系统就是为了能让学习者更好的学习计算机组成原理这门课程而准备的，平台中有大量的专业视频讲解，使得学习者在空闲时间也可以随时随地的观看视频。为了让学生更好的理解理论知识，训练学生自主思考和动手能力，视频配有一系列的习题供学习者巩固所学知识，这将会让学习者更加充分的深入了解对应课程的内容，使得学习者可以拥有身临其境的感受，提高了学习的积极性。

计算机组成原理课程培训学习平台的设计就是为了解决这样的实际困难。通过该平台，学生可以自主上网站预习或者复习，在课堂上没吸收的知识也可以在课后做到很好的消化，教师也可以根据学生在平台的学习情况及时得到反馈[1]。

## 1.2 研究目的

本研究主要目的在于：

1. 通过计算机组成原理培训平台的网络学习方法，可以拓宽学生们学习计算机组成原理课程的来源，给学生们多一种学习方式，也通过本平台的网络教育，旨在激发学生们对本课程的学习兴趣，提高学生自我学习能力，提高积极性。
2. 打破传统的线下教育，在线上也能很好的掌握课程知识点以及重点难点，学生在线下课堂中遇到较难的知识点但又未吃透，可以课后来到平台中反复观看视频，更好的掌握知识点及考点，学生能在平台上进行课前预习，课后复习，极大的提高了学生们对本课程学习的积极性，也提高了学习的自主性。教师也可以在课后对学生的学习情况进行一定的跟踪和得到一定反馈，课堂上未讲解的知识点也可以利用平台中的视频功能，让学生们在课后观看，对知识反复吸收。可以通过习题给学生们巩固课堂中学习过的知识点。

# 2 设计思路

## 2.1系统框架

前端部分采用HTML、CSS、JavaScript以及JQuery技术编写再予以Boostrap辅助优化界面美观，后端部分采用SpringBoot、MyBatis Plus和SpringSecurity框架，集成了第三方配置，搭建速度和开发更方便。数据存储方面采用MySQL和MongoDB数据库。

## 2.2开发工具以及环境

使用Intellij IDEA 开发工具开发，图形化数据库管理数据软件使用 Navicat Premium 15，使用PostMan和Apifox做接口测试，使用Git、Maven做版本控制管理。

# 3 系统设计

## 3.1 体系结构（后端部分）

如图所示，计算机组成原理培训学习平台的后端部分由以上六大模块为主要组成部分，其中用户模块中包含安全模块。邮件模块包含验证码模块，习题组模块包含选择题、判断题和主观题三大模块，视频模块中包含视频历史记录模块。可以看出，每个模块都有自己的功能，若是想增添功能以及模块，不需要对整个系统结构进行改动，在原先的基础上增添即可。如下图3.1所示。

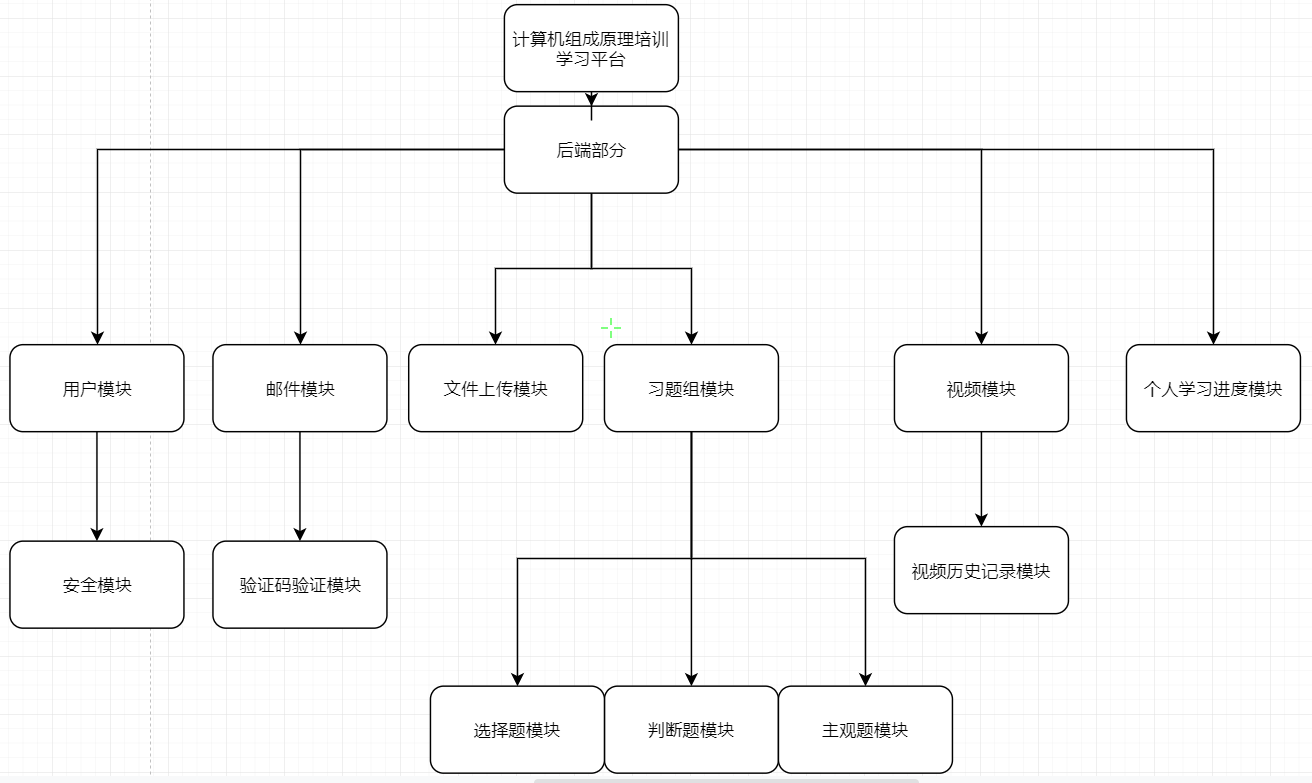
插图：居中排，图标题：五号，黑体，居中排，“图号”用Time New Roman，**图和图标题为一个整体，须在同一页**；插图与正文在同一页，则需空一行）；

图 3-1

### 3.1.1 用户模块

用户的主要功能登录以及注册，登录校验中会检查用户密码是否正确以及账号是否存在，登录成功之后，用户个人账号信息会暂时存在Redis缓存之中，并且返回一个token给到客户端方便用户后续的访问其他模块操作。用户发送注册请求，服务器接受注册信息，经过验证模块、昵称查重检测和邮箱查重检测之后，用户注册成功，并且将用户信息保存到数据库用户表中。登录功能核心代码实现详见附录A1。模块数据流图如下图3-2所示

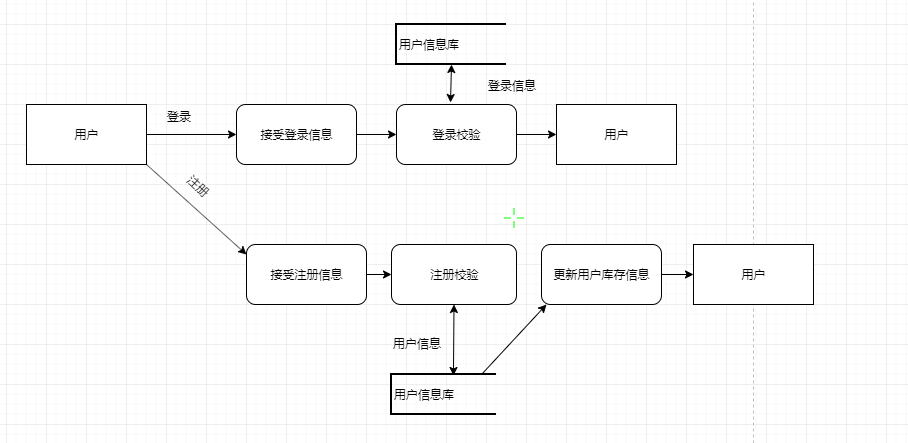


图 3‑2

### 3.1.2 安全模块

安全模块主要处理所有请求包括用户的访问请求，如果是已经登录的用户，会携带一个具有过期时间的token，访问过滤器将会拦截本次访问，并且如果携带token，将会经过处理和与服务器中的缓存做对比，得到用户基本信息，在查询该用户所拥有的权限，将合并之后的信息提供给服务端。遇到了跨域请求时，跨域过滤器将会对本次请求做一定处理，将结果由服务端发送回客户端。

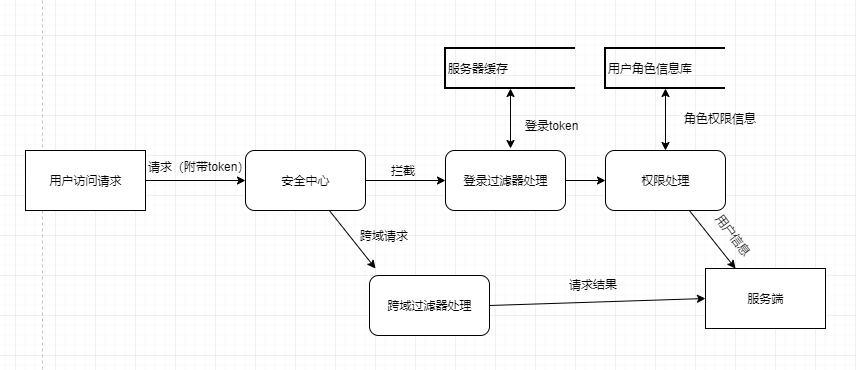


图 3-3

### 3.1.3 邮件模块

邮件模块主要分为三大块，首先是在接收了验证码请求之后，接收来自客户端的邮箱号码信息，对邮箱号码进行基本验证，通过验证之后会生成验证码，并且服务器会缓存本次验证码信息，并设置一个过期时间，最后将验证码返回给客户端，也就是向客户邮箱发送验证码邮件。

在用户填写并发送验证码给服务端之后，服务端会进行验证码校验，将客户发送的验证码与本地服务器缓存中的验证码信息进行对比，最后将结果返回给客户端。

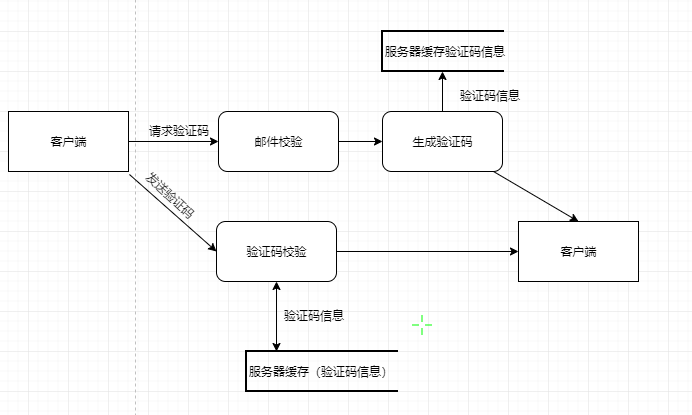


图 3‑4

### 3.1.4 文件上传模块

客户端将分片文件向服务端发送时，需携带token，经过身份认定之后，服务端接受分片文件，使用工具类对分片文件进行解析，并将分片文件临时存储在服务器磁盘中的临时分区，同时对每个分片文件进行基本信息校验，如当前分片数，文件名称，文件大小等。当最后一个分片文件到达时，服务端将开始对所有分片文件进行整合，使用缓冲输出流将每个小分片文件写入目标磁盘中，组合还原成原本的文件，同时组合好的视频文件会进行截取封面的流程，将视频的封面信息提取出来，保存到视频数据库中，并将文件以及图片信息返回给客户端。

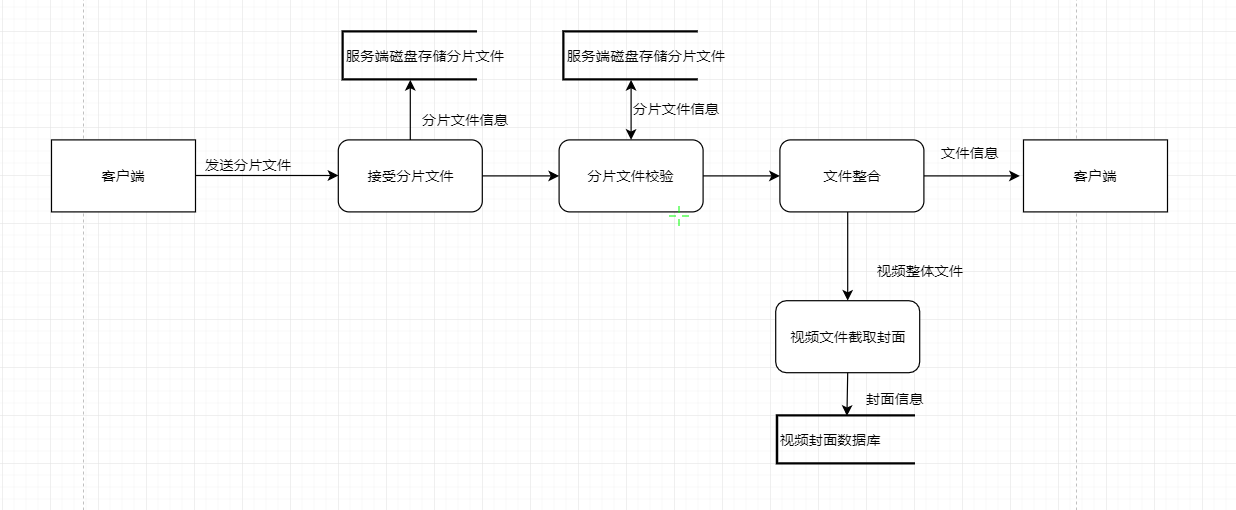
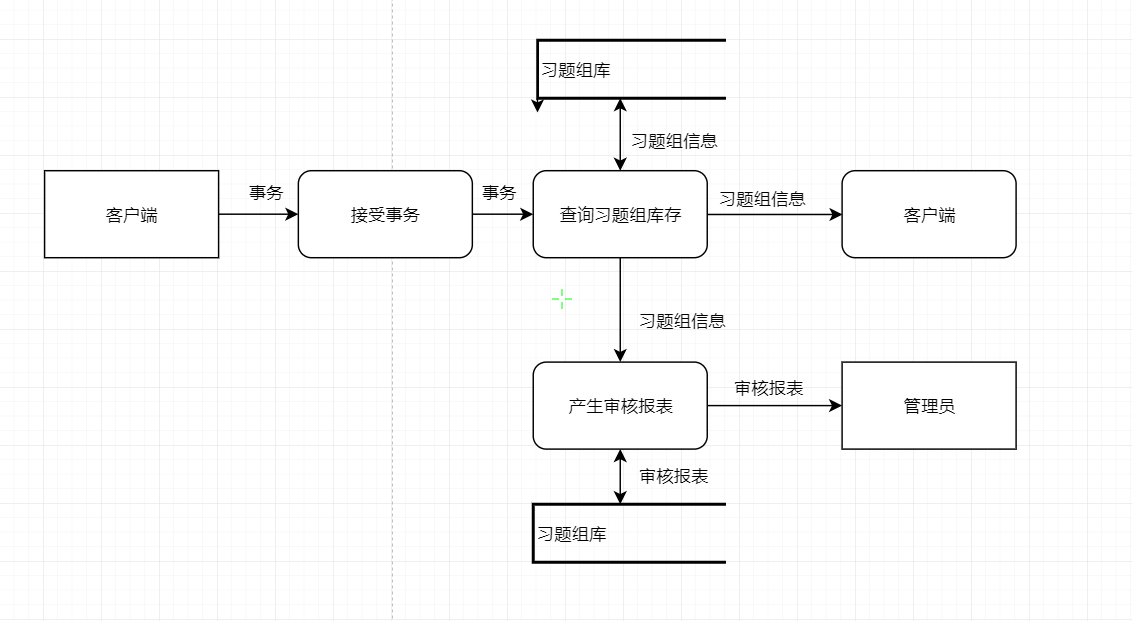


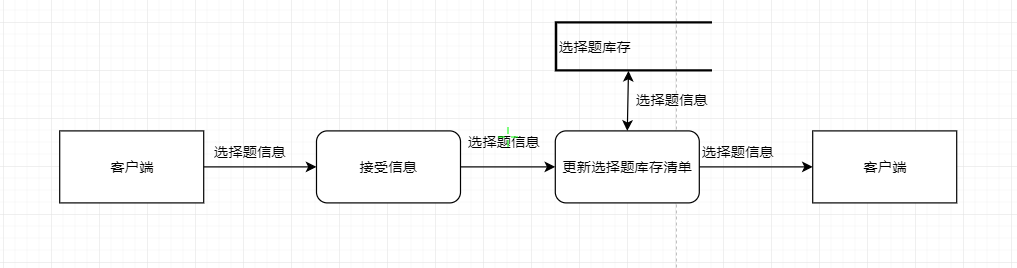
图 3‑5

### 3.1.5 习题组模块

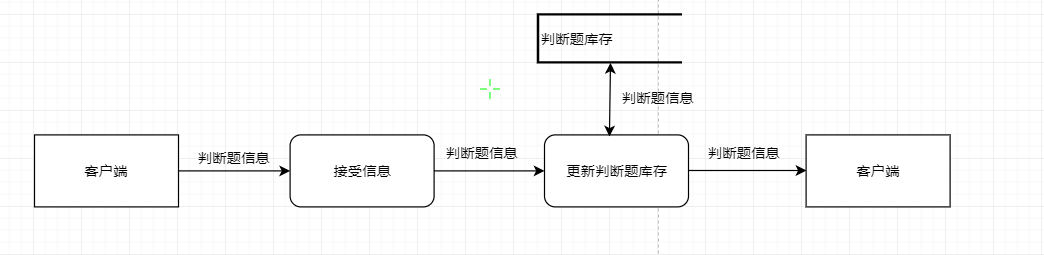
习题组模块中包含选择题、判断题和主观题三个模块，客户端向服务端发送事务请求，如新增或是查询习题组。服务端接受事务后，如果是新增习题组请求，则会将即将要新增的习题组中的选择题、判断题和主观题id集合分别的对数据库进行信息查询，校验。校验完成后将本习题组信息存到数据库中，并且产生审核报表，将审核报表存入数据库中，并且将报表发给管理员端。若是查询请求，则按习题组id去数据库查询，若是已经审核通过的习题组信息才可发向客户端。



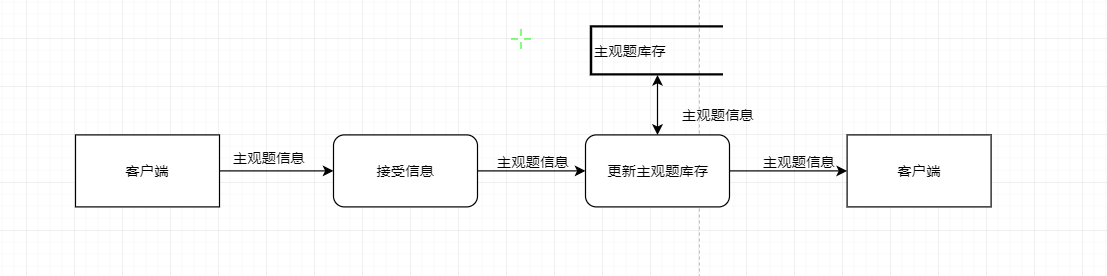
### 3.1.6 选择题模块



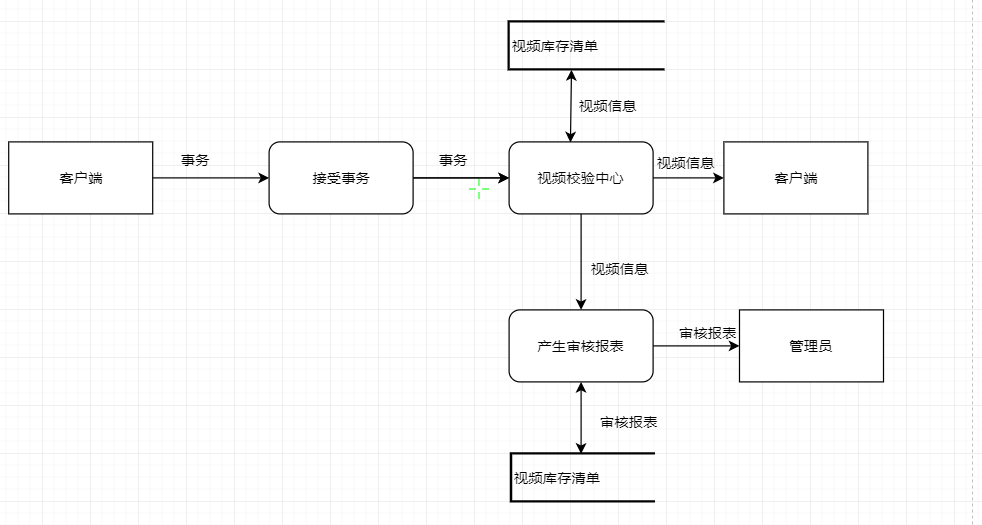
### 3.1.7 判断题模块



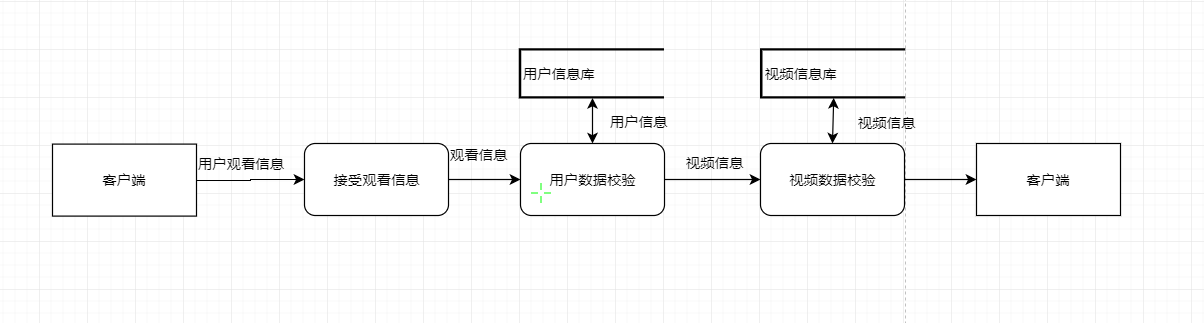
### 3.1.8 主观题模块



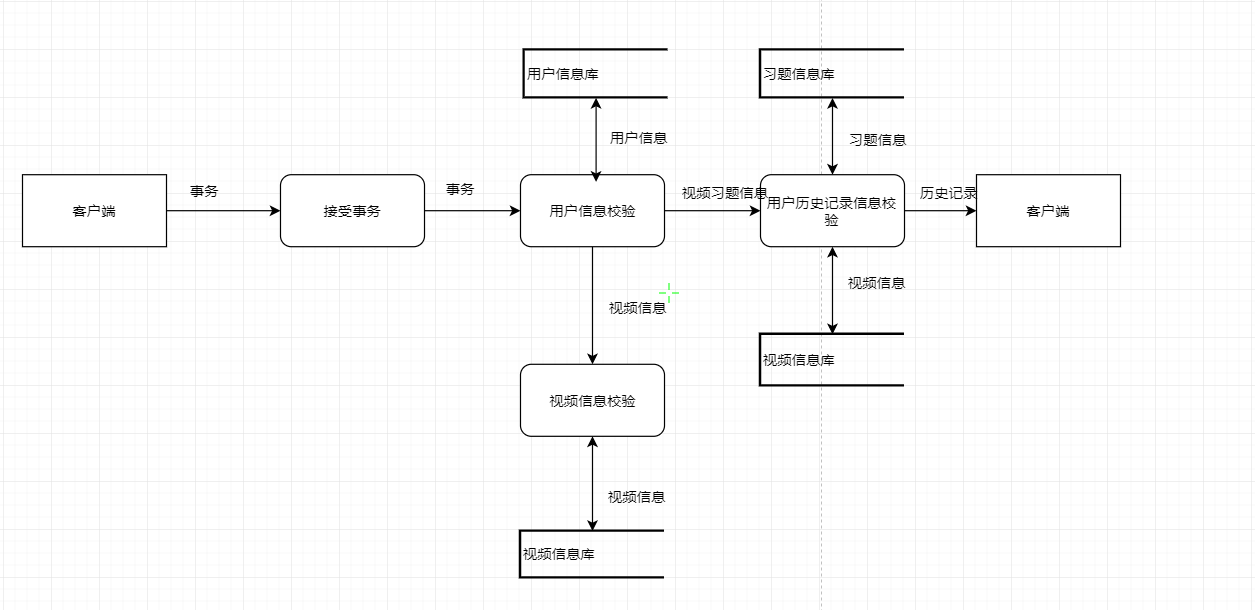
### 3.1.9 视频模块



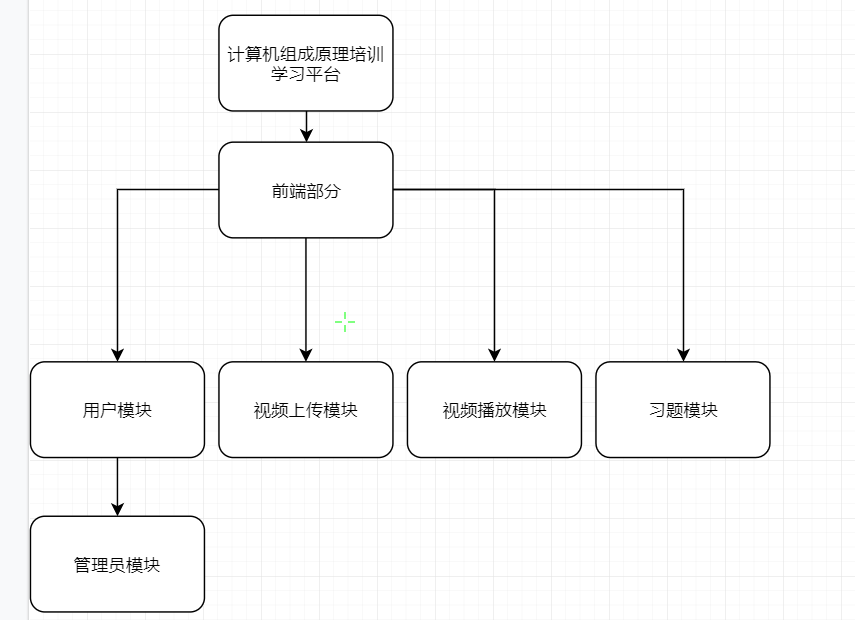
### 3.1.10 视频历史记录模块



### 3.1.11 个人学习进度模块



## 3.2 体系结构（前端部分）

而前端部分一共由四大模块组成，其中用户模块包含了管理员模块，每个模块都有对应的功能以及实现，主要是为了更好的与后端模块对接。

### 3.2.1 用户模块（含管理员模块）

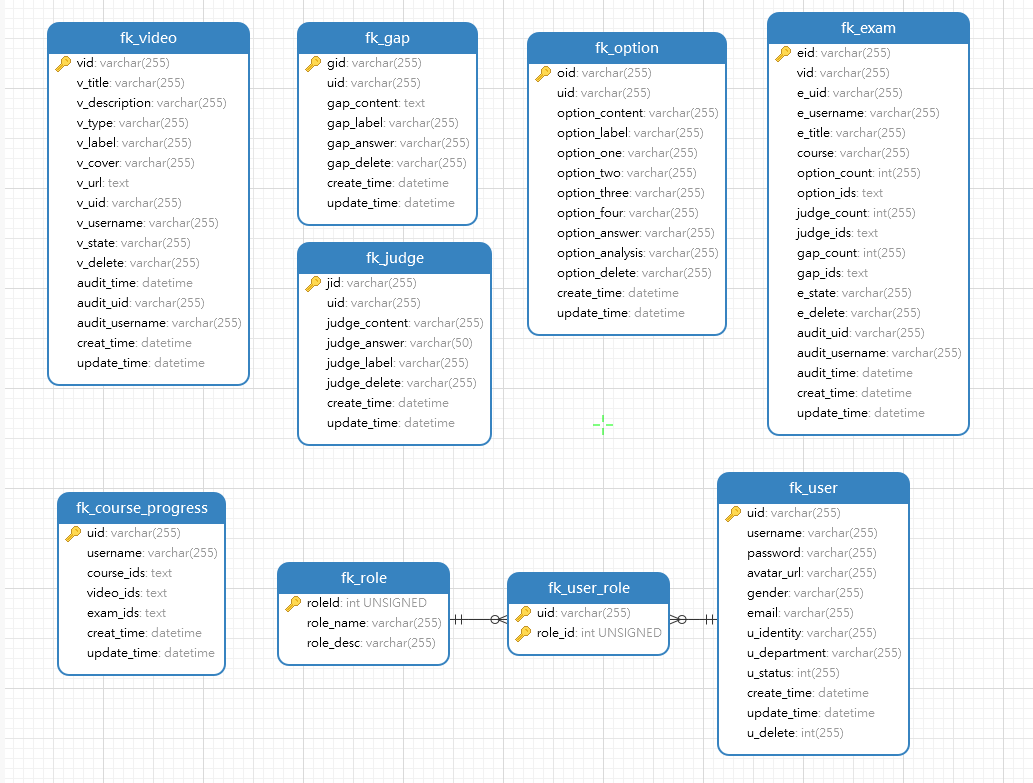
### 3.2.2 视频上传模块

### 3.2.3 视频播放模块

### 3.2.4 习题模块

# 4 数据库设计

文字介绍



## 4.1 用户表

\*\*\*\*\*\*\*\*实验结果见表6-1。（表6-1表示第6章第1序表，采用三线表边框为1磅，表格为0.5磅，表格：居中排, 标题应写在表格上方正中，不加标点，序号写在标题左方，后空一格，表文使用五号宋体，表格与正文在同一页，则需空一行，若表或图在一页的最后，则下面不需要空一行，若表或图在一页的最上方，则上面不需要空一行）

## 4.2 角色表

## 4.3 用户角色表

## 4.4 视频表

## 4.5 视频历史记录时间表

## 4.6 视频播放量表

## 4.7 习题组表

## 4.8 选择题表

## 4.9 判断题表

## 4.10 主观题表

## 4.11 个人学习进度表

# 5 系统实现

## 5.1 登录以及注册实现

## 5.2

# 6 结论

……..

参考文献:（段前24磅，段后12磅；四号黑体，顶格排。）

1. 伍超平. 基于教案-习题-视频三循环交互的在线学习平台设计[D].天津师范大学,2020.DOI:10.27363/d.cnki.gtsfu.2020.000150.辛希孟, 陈绍业, 王凤翥. 信息技术与信息服务研讨[A]. 杨洋. 中华国际论文集 [C]. 北京: 中国社会科学出版社, 1994: 12-18.
2. **注：序号采用数字加[ ]且自动排序**；**文献内容**：小四号宋体，行间距20磅，采用“英文标点”+“空格”分隔；数字及字母采用:Time New Roman其它文献类型的具体格式，请参见《肇庆学院本科毕业（设计）写作规范与印刷规范》中的（十）参考文献部分；
3. 要求近5年文献，中文≥10篇，英文文献≥2篇》

致 谢

感谢肇庆学院四年来对我的辛勤培育！感谢教师和同学，教师孜孜不倦的教导和同学的友爱互助。

本论文是在我的指导老师的亲切关怀和悉心指导下完成的。从课题的选择到项目的最终完成，XXX老师都始终给予我细心的指导和不懈的支持。在此谨向XXX老师致以诚挚的谢意和崇高的敬意！

注：1、“致谢”两字间隔为两格，四号黑体，居中排，段前段后为24磅；2、致谢内容：小四号宋体，行间距20磅，内容长度限一页内；3、本页独立成页，单面打印）

附录A（单独成页，四号黑体，居中排，必要时采用，可选）

附A1

登录模块

@Override  
*public Authentication* attemptAuthentication ( *HttpServletRequest* request, *HttpServletResponse* response ) *throws* AuthenticationException  
{  
 *try* {  
 User user = *new* ObjectMapper().readValue(request.getInputStream(), User.*class*);  
 UsernamePasswordAuthenticationToken authRequest = *new* UsernamePasswordAuthenticationToken(user.getEmail(),user.getPassword());  
 *return this*.getAuthenticationManager().authenticate(authRequest);  
 }  
 *catch* ( Exception e )  
 {  
 *try* {  
 response.setContentType("application/json;charset=utf-8");  
 response.setStatus(*HttpServletResponse*.*SC\_UNAUTHORIZED*);  
 *Map*<Object, Object> resultMap = *new* HashMap<>(16);  
 resultMap.put("code",*HttpServletResponse*.*SC\_UNAUTHORIZED*);  
 resultMap.put("msg","用户名或密码错误");  
 ServletOutputStream out = response.getOutputStream();  
 OutputStreamWriter ow = *new* OutputStreamWriter(out, StandardCharsets.*UTF\_8*);  
 ow.write(*new* ObjectMapper().writeValueAsString(resultMap));  
 ow.flush();  
 ow.close();  
 }  
 *catch* ( Exception ex )  
 {  
 *throw new* RuntimeException(ex);  
 }  
 *throw new* RuntimeException(e);  
 }  
}

*/\*\*  
 \* 认证成功回写token  
 \*/*@Override  
*public void* successfulAuthentication ( *HttpServletRequest* request, *HttpServletResponse* response, *FilterChain* chain, *Authentication* authResult )  
{  
 User user = *new* User();  
 user.setUsername(authResult.getName());  
 user.setRoles((*List*<Role>) authResult.getAuthorities());  
 *//一天的时间* String token = JwtUtils.*generateTokenExpireInMinutes*(user, rsaKeyConfig.getPrivateKey());  
 *//将token写入redis缓存，需要查询一次数据库* LambdaQueryWrapper<User> lqw = *new* LambdaQueryWrapper<>();  
 lqw.eq(User::getUsername, user.getUsername());  
 User selectOne = userService.getOne(lqw);  
 UserDTO userDTO = BeanUtil.*copyProperties*(selectOne, UserDTO.*class*);  
 *List*<Role> roles = roleMapper.findUserRole(userDTO.getUid());  
 userDTO.setRoles(roles);  
 UserHolder.*saveUser*(userDTO);  
 *Map*<String, Object> userMap = BeanUtil.*beanToMap*(userDTO, *new* HashMap<>(16),  
 CopyOptions.*create*()  
 .setIgnoreNullValue(*true*)  
 .setFieldValueEditor(( fieldName, fieldValue ) -> String.*valueOf*(fieldValue)));  
 *//存入redis并设置token有效期,登录后还需要更新token* String tokenKey = RedisConstants.*LOGIN\_USER\_KEY* + token;  
 response.addHeader("authorization", "Bearer " + token);  
 stringRedisTemplate.opsForHash().putAll(tokenKey, userMap);  
 stringRedisTemplate.expire(tokenKey, RedisConstants.*LOGIN\_USER\_TTL*, TimeUnit.*MINUTES*);  
 *try* {  
 response.setContentType("application/json;charset=utf-8");  
 response.setStatus(*HttpServletResponse*.*SC\_OK*);  
 PrintWriter writer = response.getWriter();  
 *Map*<Object, Object> resultMap = *new* HashMap<>(16);  
 resultMap.put("code", *HttpServletResponse*.*SC\_OK*);  
 resultMap.put("user", userDTO);  
 resultMap.put("msg", "认证通过");  
 writer.write(*new* ObjectMapper().writeValueAsString(resultMap));  
 writer.flush();  
 writer.close();  
 }  
 *catch* ( IOException ex )  
 {  
 *throw new* RuntimeException(ex);  
 }  
}

附录内容采用双面打印，附录图表序号表示方法参见《附件1计算机科学与软件学院本科生毕业论文（设计）写作与印制规范中的（十一）附录部分。

对于一些不宜放在正文中的重要支撑材料，可编入毕业论文的附录中。包括与论文有关的图表、计算机程序、运行结果、主要设备、仪器仪表的性能指标和测试精度等，附录的篇幅不宜太多。

特别提醒：

（模板中的所有红色、蓝色字体为格式说明，请删除！！！）更多详情，参见：《计算机科学与软件学院本科生毕业论文（设计）写作与印制规范（2021年10月）》（附件6）