

小济在线考试系统 – 实验报告

- 学号：1951976
- 姓名：李林飞

小济在线考试系统 – 实验报告

1. 实验要求
2. 系统设计实现
 - 2.1 数据库的设计
 - 2.1.1 数据库ER图设计
 - 2.1.2 数据库表设计
 - 2.2 后端设计与实现
 - 2.2.1 项目配置
 - 2.2.2 业务逻辑 (以在线评分为例)
 - 2.2.3 Swagger接口文档
 - 2.3 前端设计与实现
 - 2.3.1 登录与注册
 - 2.3.2 使用Cookies保存用户信息
 - 2.3.3 在线考试系统
 1. 学生界面
 2. 管理员界面
 3. 教师界面
3. 系统运行
 - 3.1 前端运行
 - 3.2 后端运行
4. 实验总结

1. 实验要求

• 功能1

开发用户注册和登录模块，不同的用户对数据库有不同的操作权限（浏览数据库和编辑数据库）。
(提示：在注册时将有关信息保存到数据库中；当再次登录时，根据用户输入的用户名和密码在数据库中查找，不同的用户给出不同的页面进行操作。)

• 功能2

开发一个简单的在线考试程序，可以包括若干道单选题、多选题，单击交卷按钮后就可以根据标准答案在线评分。

• 功能3

开发一个页面，当客户第一次访问时，需在线注册姓名、性别等信息，然后将信息保存到Cookies中，当下次如该用户再访问，则显示“某某，您好，您是第几次光临本站”的欢迎信息。

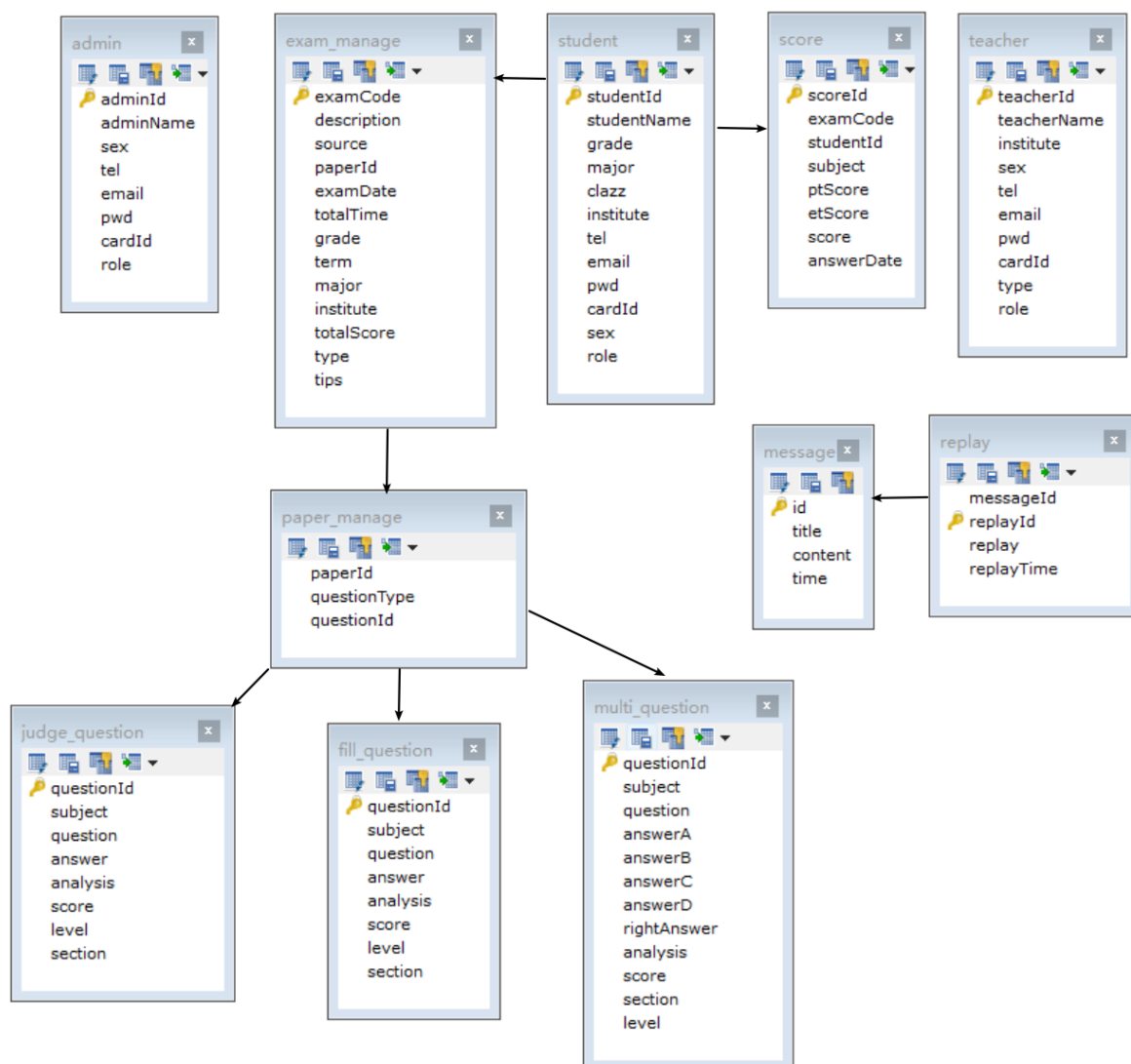
2. 系统设计实现

2.1 数据库的设计

2.1.1 数据库ER图设计

数据库设计比较简单，没有严格遵守第三范式。为了简化问题，对传统的线上考试系统做了以下简化：

- 所有的考试题都是共有的，相当于一个题库，老师是题库的提供者，但不与试题进行关联。
- 学生是刷题者，与老师没有直接的关联关系。本身具有分数、参与哪次考试的关联信息。
- 一场考试对应一张试卷，一张试卷有三种题型组成：选择题、判断题、填空题。
- 超级管理员与具体实体类没有关联关系，只是在权限上提供更多操作。
- 学生留言区提供自由式评论与留言，不记录留言和发表评论的人。
- 不记录学生考试选择的选项，每个试题中有答案，学生作答后在前端计算总分。



2.1.2 数据库表设计

- 管理员信息表

Field	Type	Collation	Null	Key	Default	Comment
adminId	int(9)	(NULL)	NO	PRI	(NULL)	ID号
adminName	varchar(20)	utf8_bin	YES		(NULL)	姓名
sex	varchar(2)	utf8_bin	YES	MUL	(NULL)	性别
tel	varchar(11)	utf8_bin	YES		(NULL)	电话号码
email	varchar(20)	utf8_bin	YES		(NULL)	电子邮箱
pwd	varchar(16)	utf8_bin	YES		(NULL)	密码
cardId	varchar(18)	utf8_bin	YES		(NULL)	身份证号
role	varchar(1)	utf8_bin	YES		(NULL)	角色(0管理员, 1教师, 2学生)

- 考试管理表

Field	Type	Collation	Null	Key	Default	Comment
examCode	int(9)	(NULL)	NO	PRI	(NULL)	考试编号
description	varchar(50)	utf8_bin	YES		(NULL)	该次考试介绍
source	varchar(20)	utf8_bin	YES		(NULL)	课程名称
paperId	int(10)	(NULL)	YES		(NULL)	试卷编号
examDate	varchar(10)	utf8_bin	YES		(NULL)	考试日期
totalTime	int(3)	(NULL)	YES		(NULL)	持续时长
grade	varchar(10)	utf8_bin	YES		(NULL)	年级
term	varchar(10)	utf8_bin	YES		(NULL)	学期
major	varchar(20)	utf8_bin	YES		(NULL)	专业
institute	varchar(20)	utf8_bin	YES		(NULL)	学院
totalScore	int(4)	(NULL)	YES		(NULL)	总分
type	varchar(255)	utf8_bin	YES		(NULL)	考试类型
tips	varchar(255)	utf8_bin	YES		(NULL)	考生须知

- 填空题表

Field	Type	Collation	Null	Key	Default	Comment
questionId	int(9)	(NULL)	NO	PRI	(NULL)	试题编号
subject	varchar(20)	utf8_bin	YES		(NULL)	考试科目
question	varchar(255)	utf8_bin	YES		(NULL)	试题内容
answer	varchar(255)	utf8_bin	YES		(NULL)	正确答案
analysis	varchar(255)	utf8_bin	YES		(NULL)	题目解析
score	int(2)	(NULL)	YES		2	分数
level	varchar(5)	utf8_bin	YES		(NULL)	难度等级
section	varchar(20)	utf8_bin	YES		(NULL)	所属章节

- 判断题表

Field	Type	Collation	Null	Key	Default	Comment
questionId	int(9)	(NULL)	NO	PRI	(NULL)	试题编号
subject	varchar(20)	utf8_bin	YES		(NULL)	考试科目
question	varchar(255)	utf8_bin	YES		(NULL)	试题内容
answer	varchar(255)	utf8_bin	YES		(NULL)	正确答案
analysis	varchar(255)	utf8_bin	YES		(NULL)	题目解析
score	int(2)	(NULL)	YES		2	分数
level	varchar(1)	utf8_bin	YES		(NULL)	难度等级
section	varchar(20)	utf8_bin	YES		(NULL)	所属章节

- 留言表

Field	Type	Collation	Null	Key	Default	Comment
questionId	int(9)	(NULL)	NO	PRI	(NULL)	试题编号
subject	varchar(20)	utf8_bin	YES		(NULL)	考试科目
question	varchar(255)	utf8_bin	YES		(NULL)	试题内容
answer	varchar(255)	utf8_bin	YES		(NULL)	正确答案
analysis	varchar(255)	utf8_bin	YES		(NULL)	题目解析
score	int(2)	(NULL)	YES		2	分数
level	varchar(1)	utf8_bin	YES		(NULL)	难度等级
section	varchar(20)	utf8_bin	YES		(NULL)	所属章节

- 评论回复表

Field	Type	Collation	Null	Key	Default	Comment
paperId	int(9)	(NULL)	YES		(NULL)	试卷编号
questionType	int(1)	(NULL)	YES		(NULL)	题目类型
questionId	int(9)	(NULL)	YES		(NULL)	题目编号

- 选择题表

Field	Type	Collation	Null	Key	Default	Comment
questionId	int(9)	(NULL)	NO	PRI	(NULL)	试题编号
subject	varchar(20)	utf8_bin	YES		(NULL)	考试科目
question	varchar(255)	utf8_bin	YES		(NULL)	问题题目
answerA	varchar(255)	utf8_bin	YES		(NULL)	选项A
answerB	varchar(255)	utf8_bin	YES		(NULL)	选项B
answerC	varchar(255)	utf8_bin	YES		(NULL)	选项C
answerD	varchar(255)	utf8_bin	YES		(NULL)	选项D
rightAnswer	varchar(10)	utf8_bin	YES		(NULL)	正确答案
analysis	varchar(255)	utf8_bin	YES		(NULL)	题目解析
score	int(2)	(NULL)	YES		2	分数
section	varchar(20)	utf8_bin	YES		(NULL)	所属章节

Field	Type	Collation	Null	Key	Default	Comment
level	varchar(1)	utf8_bin	YES		(NULL)	难度等级

- 试卷表

Field	Type	Collation	Null	Key	Default	Comment
paperId	int(9)	(NULL)	YES		(NULL)	试卷编号
questionType	int(1)	(NULL)	YES		(NULL)	题目类型
questionId	int(9)	(NULL)	YES		(NULL)	题目编号

- 成绩表

Field	Type	Collation	Null	Key	Default	Comment
scoreId	int(9)	(NULL)	NO	PRI	(NULL)	分数编号
examCode	int(9)	(NULL)	YES		(NULL)	考试编号
studentId	int(9)	(NULL)	YES		(NULL)	学号
subject	varchar(20)	utf8_bin	NO		(NULL)	课程名称
ptScore	int(5)	(NULL)	YES		(NULL)	平时成绩
etScore	int(5)	(NULL)	YES		(NULL)	期末成绩
score	int(5)	(NULL)	YES		(NULL)	总成绩
answerDate	varchar(10)	utf8_bin	YES		(NULL)	答题日期

- 学生信息表

Field	Type	Collation	Null	Key	Default	Comment
studentId	int(9)	(NULL)	NO	PRI	(NULL)	ID
studentName	varchar(20)	utf8_bin	YES		(NULL)	姓名
grade	varchar(4)	utf8_bin	YES		(NULL)	年级
major	varchar(20)	utf8_bin	YES		(NULL)	专业
clazz	varchar(10)	utf8_bin	YES		(NULL)	班级
institute	varchar(30)	utf8_bin	YES		(NULL)	学院
tel	varchar(11)	utf8_bin	YES		(NULL)	电话号码
email	varchar(30)	utf8_bin	YES		(NULL)	电子邮件
pwd	varchar(16)	utf8_bin	YES		(NULL)	密码
cardId	varchar(18)	utf8_bin	YES		(NULL)	身份证号
sex	varchar(2)	utf8_bin	YES		(NULL)	性别
role	varchar(1)	utf8_bin	YES		2	角色(0管理员, 1教师, 2学生)

- 教师信息表

teacherId	int(9)	(NULL)	NO	PRI	(NULL)	ID
teacherName	varchar(20)	utf8_bin	YES		(NULL)	姓名
institute	varchar(20)	utf8_bin	YES		(NULL)	学院
sex	varchar(2)	utf8_bin	YES		(NULL)	性别
tel	varchar(11)	utf8_bin	YES		(NULL)	电话号码
email	varchar(20)	utf8_bin	YES		(NULL)	邮箱
pwd	varchar(16)	utf8_bin	YES		(NULL)	密码
cardId	varchar(18)	utf8_bin	YES		(NULL)	身份证号
type	varchar(20)	utf8_bin	YES		(NULL)	职称
role	varchar(1)	utf8_bin	YES		(NULL)	角色 (0管理员, 1教师, 2学生)

2.2 后端设计与实现

后端主要使用SpringBoot+MyBatisPLUS+MySQL实现。业务逻辑大都是增删改查，并不复杂。集成Swagger3.0创建接口文档。技术栈：

- SpringBoot 2.6.7
- MyBatisPLUS:
- MySQL: 5.7
- Swagger: 3.0
- Java: 1.8

2.2.1 项目配置

- 数据库连接配置

```
1 # 数据库配置 MySQL 5.7
2 spring.datasource.username=root
3 spring.datasource.password=123456
4 spring.datasource.url=jdbc:mysql://localhost:3306/xiaoji_exam?
  useUnicode=true&characterEncoding=UTF-8&useSSL=false&serverTimezone=UTC
5 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
6 spring.datasource.type=com.alibaba.druid.pool.DruidDataSource
```

- 配置MyBatisPLUS做分页查询

```
1 @EnableTransactionManagement
2 @Configuration
3 @MapperScan("com.exam.mapper")
4 public class MybatisPlusConfig {
5     /**
6      * 新的分页插件,一缓和二缓遵循mybatis的规则,需要设置
7      * MybatisConfiguration#useDeprecatedExecutor = false 避免缓存出现问题(该属
8      * 性会在旧插件移除后一同移除)
9      */
10    @Bean
11    public MybatisPlusInterceptor mybatisPlusInterceptor() {
12        MybatisPlusInterceptor interceptor = new
13        MybatisPlusInterceptor();
14        interceptor.addInnerInterceptor(new
15        PaginationInnerInterceptor(DbType.H2));
16        return interceptor;
17    }
18 }
```

- 配置Swagger接口文档: 访问地址: <http://localhost:8080/swagger-ui/index.html>

```
1 @Configuration
2 @EnableOpenApi
3 public class SwaggerConfig {
4     @Bean
```



```

5     public Docket docket(){
6         return new Docket(DocumentationType.OAS_30)
7             .apiInfo(apiInfo()).enable(true)
8             .select()
9             //apis: 添加swagger接口提取范围
10            .apis(RequestHandlerSelectors.basePackage("com.exam"))
11
12            //apis(RequestHandlerSelectors.withMethodAnnotation(ApiOperation.class))
13
14            .paths(PathSelectors.any())
15            .build();
16    }
17
18    private ApiInfo apiInfo(){
19        return new ApiInfoBuilder()
20            .title("小济线上考试项目接口文档")
21            .description("基于SpringBoot的在线考试系统")
22            .contact(new Contact("1951976 李林飞",
23                "https://github.com/lif314", "1951976@tongji.edu.cn"))
24            .version("1.0")
25            .build();
26    }
27 }

```

2.2.2 业务逻辑 (以在线评分为例)

- 当学生查询在考试页面中点击某一场考试：后端在controller中接口查询试卷试题的请求。由于一套试卷是由不同数量的选择题、填空题和判断题构成，所以通过关联关系查询到具体的试题信息。试题信息中包含题目本身以及试题答案。

```

1     @ApiOperation("根据试卷id查询试卷信息")
2     @GetMapping("/paper/{paperId}")
3     public Map<Integer, List<?>> findById(@PathVariable("paperId")
4     Integer paperId) {
5         List<MultiQuestion> multiQuestionRes =
6         multiQuestionService.findByIdAndType(paperId); //选择题题库 1
7         List<FillQuestion> fillQuestionsRes =
8         fillQuestionService.findByIdAndType(paperId); //填空题题库 2
9         List<JudgeQuestion> judgeQuestionRes =
10        judgeQuestionService.findByIdAndType(paperId); //判断题题库 3
11        Map<Integer, List<?>> map = new HashMap<>();
12        map.put(1,multiQuestionRes);
13        map.put(2,fillQuestionsRes);
14        map.put(3,judgeQuestionRes);
15        return map;
16    }

```

- 前端接受到试题信息渲染在页面上，然后通过比对用户输入与试题的答案进行评分。并实时显示考试总分。
- 当接受考试时，前端向后端发送添加考试成绩信息的请求

```

1      @ApiOperation("添加学生成绩")
2      @PostMapping("/score")
3      public ApiResult add(@RequestBody Score score) {
4          int res = scoreService.add(score);
5          if (res == 0) {
6              return ApiResultHandler.buildApiResult(400,"成绩添加失
败",res);
7          }else {
8              return ApiResultHandler.buildApiResult(200,"成绩添加成
功",res);
9          }
10     }

```

- 该请求在controller中接受，在service中处理

```

1      @Override
2      public int add(Score score) {
3          return scoreMapper.add(score);
4      }

```

- 通过MyBatisPLUS，会映射到一个Mapper上

```

1      /**
2       * @param score 添加一条成绩记录
3       */
4      @Options(useGeneratedKeys = true,keyProperty = "scoreId")
5      @Insert("insert into
score(examCode,studentId,subject,ptScore,etScore,score,answerDate)
values(#{examCode},#{studentId},#{subject},#{ptScore},#{etScore},#
{score},#{answerDate})")
6      int add(Score score);

```

- MyBatisPLUS根据mapper与数据库进行连接，并完成注解上的操作。

其余业务逻辑都差不多。同时，MybatisPLUS提供了分页查询机制。

2.2.3 Swagger接口文档

Servers

http://localhost:8080 - Inferred Url

判断题接口	Judge Question Controller	>
填空题接口	Fill Question Controller	>
学生管理接口	Student Controller	>
成绩管理接口	Score Controller	>
用户登录	Login Controller	>
留言管理接口	Message Controller	>
留言评论接口	Replay Controller	>
管理员系统接口	Admin Controller	>

- 栗子【考试管理接口】

考试管理接口		Exam Manage Controller	▼
PUT	/exam	根据ID更新试卷	
POST	/exam	添加试卷	
GET	/exam/{examCode}	根据ID查找试卷	
DELETE	/exam/{examCode}	根据ID删除试卷	
GET	/examManagePaperId	根据试卷编号查询考试信息	
GET	/exams	不分页查询所有试卷	
GET	/exams/{page}/{size}	分页查询所有试卷	

2.3 前端设计与实现

前端主要采用Vue和ElemetUI进行开发，完成项目要求。同时扩展了一些功能，但部分界面上的功能还没有实现。

前端技术栈：

- Vue@2
- Vuex@3
- VueRouter
- VueCookies
- ElementUI
- axios
- echarts

2.3.1 登录与注册

由于是在线考试系统，以学生管理系统作为参考，学生注册功能改为由管理员发放账号(学号)。学生可以自己的密码等，但不能修改账号。

对于学生、教师和管理员，我通过实体类中的 `role` 字段进行判断：0管理员，1教师，2学生。前端根据用户输入的账号向后端发送请求获取用户的完整信息，然后根据 `role` 字段判断用户角色，然后使用VueRouter进行路由跳转，不同角色的用户在不同的页面中进行操作。

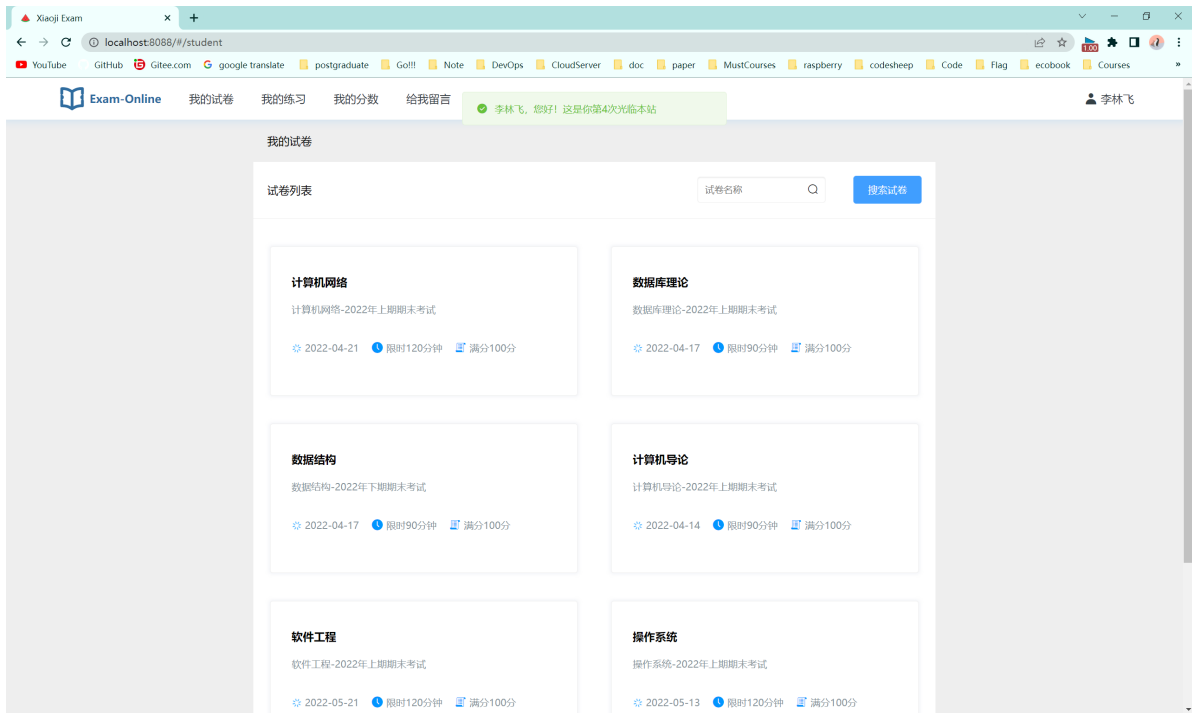
```
1  if (resData !== null) {  
2      switch (resData.role) {  
3          case "0": //管理员  
4              this.$router.push({ path: "/index" }); //跳转到首页  
5              break;  
6          case "1": //教师  
7              this.$router.push({ path: "/index" }); //跳转到教师用户  
8              break;  
9          case "2": //学生  
10             this.$router.push({ path: "/student" });  
11             break;  
12         }  
13     }
```

- 登录界面



版权所有 ©2022 同济大学软件学院李科飞 保留所有权利 沪ICP备19001371号

- 学生登录



- 管理员登录



- 教师登录

XIAOJI-EXAM

张林, 您好! 这是你第2次光临本站

考试管理

题库管理

成绩查询

学生管理

很高兴遇见你,张林老师。

教务公告:

疫情防控期间本科生教学相关业务FAQ

疫情防控期间本科生教学相关业务FAQ

疫情防控期间本科生教学相关业务FAQ

疫情防控期间本科生教学相关业务FAQ

- 管理员或教师创建学生初始账号

XIAOJI-EXAM

考试管理

题库管理

成绩查询

学生管理

学生管理

添加学生

姓名

linfei li

性别

男

学院

软件

所属专业

软件

年级

2019

班级

1

电话号码

18387400236

身份证号

12141452535353

邮箱

1212323423@qq.com

密码

123456

立即创建

取消

- 创建结果

考试管理

题库管理

成绩查询

学生管理

学生管理

添加学生

学生管理

姓名	学院	专业	年级	班级	性别	联系方式	操作
王二虎	软件工程学院	计算机科学与技术	2019	2	男	13585439532	<div>编辑删除</div>
李林飞	软件工程学院	计算机科学与技术	2019	2	男	13658377857	<div>编辑删除</div>
王文林	软件工程学院	信息工程	2020	1	男	15583829425	<div>编辑删除</div>
linfei li	软件	软件	2019	1	男	18387400236	<div>编辑删除</div>

共 4 条6条/页

<1>

前往1页

2.3.2 使用Cookies保存用户信息

- 当用户登陆时，可以使用Vue-Cookies存储用户信息

```

1  if (resData !== null) {
2      switch (resData.role) {
3          case "0": //管理员
4              this.$cookies.set("cname", resData.adminName);
5              this.$cookies.set("cid", resData.adminId);
6              this.$cookies.set("role", 0);
7              this.countLoginNum(resData.adminName);
8              this.$router.push({ path: "/index" }); //跳转到首页
9              break;
10         case "1": //教师
11             this.$cookies.set("cname", resData.teacherName);
12             this.$cookies.set("cid", resData.teacherId);
13             this.$cookies.set("role", 1);
14             this.countLoginNum(resData.teacherName);
15             this.$router.push({ path: "/index" }); //跳转到教师用户
16             break;
17         case "2": //学生
18             this.$cookies.set("cname", resData.studentName);
19             this.$cookies.set("cid", resData.studentId);
20             this.countLoginNum(resData.studentName);
21             this.$router.push({ path: "/student" });
22             break;
23     }
24 }
```

- 同时，用户登录时会在Cookie中存储用户登录的次数：如果用户第一次登录，则将用户名cname+'num'拼串作为key，以num初始值为1作为值存储在Cookie中；如果用户不是第一次登录，则通过key获取用户的登录次数，然后+1之后再存储在Cookie中。

```

1  // 统计用户登录次数
2  countLoginNum(cname) {
3      const key = cname + "num";
4      if (this.$cookies.isKey(key)) {
5          // console.log("1111s")
6          let strNum = this.$cookies.get(key);
7          console.log(strNum)
8          // 存在用户登录信息
9          let num = parseInt(strNum);
```

```

10     this.$cookies.set(key, num + 1);
11   } else {
12     this.$cookies.set(key, 1);
13   }
14   console.log(this.$cookies.get(key));
15 },

```

- 当用户登录成功时，将用户登录次数显示出来

```

1   userInfo() {
2     let studentName = this.$cookies.get("cname")
3     let studentId = this.$cookies.get("cid")
4     let loginNum = this.$cookies.get(studentName+"num");
5     // 显示登录消息
6     this.$message({
7       message: `${studentName}, 您好! 这是你第${loginNum}次光临本站`,
8       type: 'success'
9     });
10    console.log(`studentId${studentId}`)
11    console.log(`studentName ${studentName}`)
12    this.user.userName = studentName
13    this.user.studentId = studentId
14  },

```

Name	Value	Domain	Path	Expires / M...	Size	HttpOnly	Secure	SameSite	SameParty
%E5%BC%A0%E6%9E%97num	3	localhost	/	2022-05-09...	22			Lax	
%E8%B6%B5%E7%B8%A7%E7%AE%A1%E7%...	9	localhost	/	2022-05-09...	49			Lax	
%E6%9D%8E%E6%9E%97%E9%A3%9Eenum	6	localhost	/	2022-05-09...	31			Lax	

- 当用户退出登录时，清除用户信息，但在浏览器中保留登录次数信息

```

1   exit() { //退出登录
2     this.$router.push({path:"/"}) //跳转到登录页面
3     this.$cookies.remove("cname") //清除cookie
4     this.$cookies.remove("cid")
5   },

```

- 演示

Exam-Online

我的试卷

我的练习

我的分数

给我留言

李林飞, 您好! 这是你第5次光临本站

李林飞

我的试卷

试卷列表

试卷名称

Q

搜索试卷

计算机网络

计算机网络-2022年上期期末考试

2022-04-21

限时120分钟

满分100分

数据库理论

数据库理论-2022年上期期末考试

2022-04-17

限时90分钟

满分100分

数据结构

数据结构-2022年下期期末考试

2022-04-17

限时90分钟

满分100分

计算机导论

计算机导论-2022年上期期末考试

2022-04-14

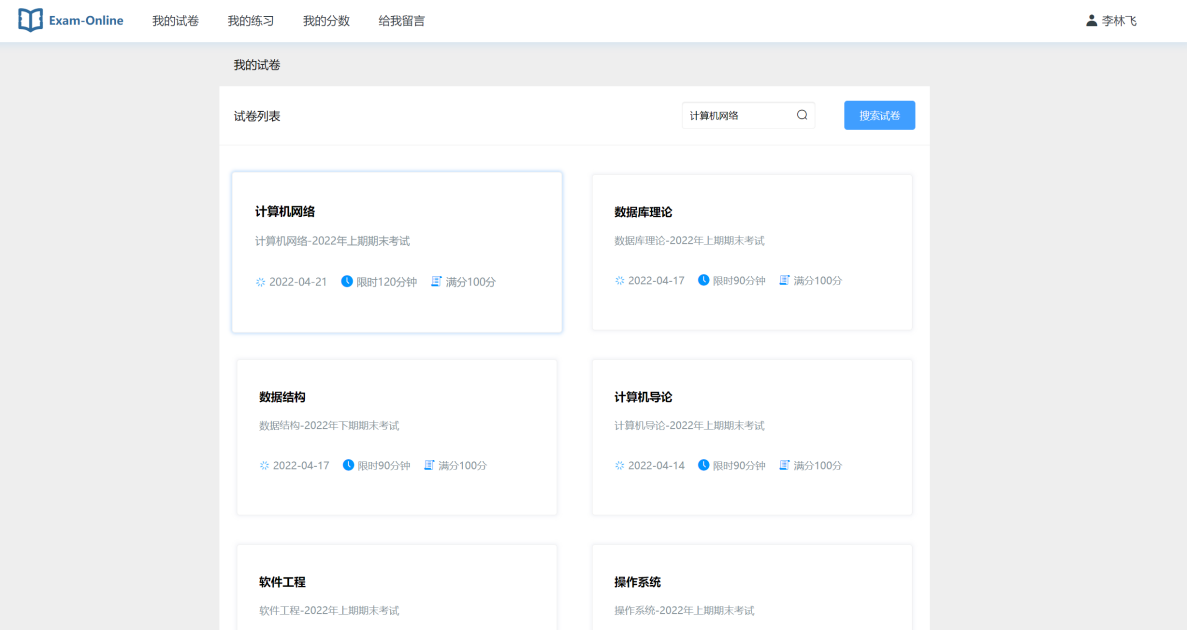
限时90分钟

满分100分

2.3.3 在线考试系统

1. 学生界面

- 【我的试卷】：查看题库中的试卷【目前只导入《计算机网络》科目的数据】



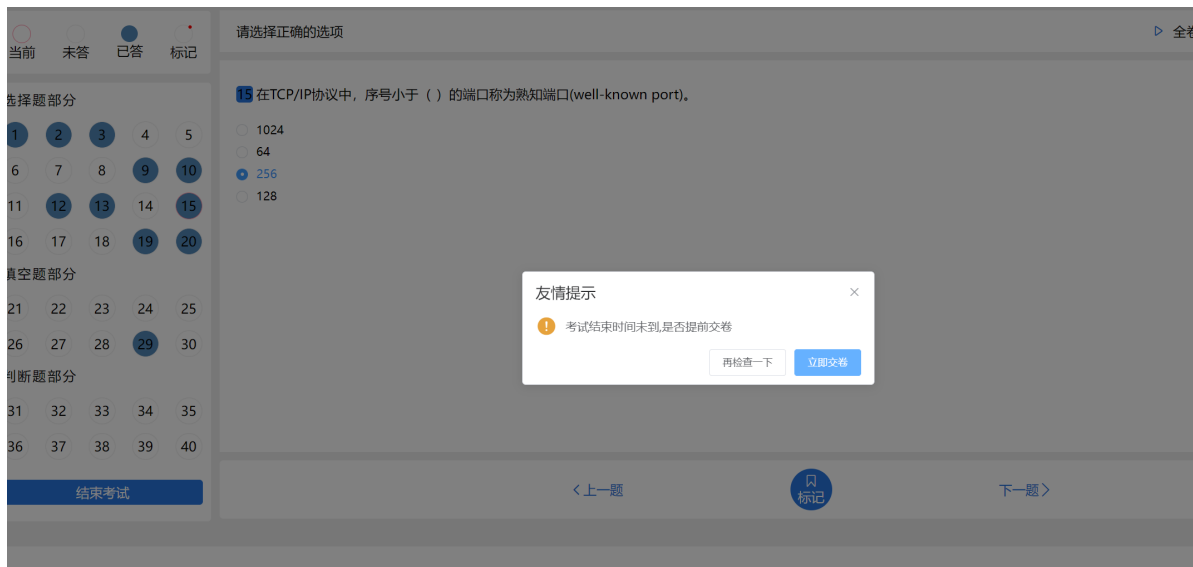
- 点击【计算机网络】进入考试界面



- 开始考试：试题由三种题型组成



- 结束考试。提交试卷



- 查看考试成绩【由于后端返回试题中已经带有试题答案，前端获取用户输入后，可以实时比对答案计算分数】



- 查询考试记录

我的分数			
考试日期	课程名称	考试分数	是否及格
2022-04-23	计算机网络	70	及格
2022-04-20	数据库理论	78	及格
2022-04-20	数据库理论	80	及格
2022-04-20	数据库理论	83	及格
2022-04-25	计算机网络	80	及格
2022-04-29	计算机网络	0	不及格
2022-04-26	计算机网络	0	不及格
2022-05-06	计算机网络	2	不及格
2022-05-06	计算机网络	10	不及格
2022-05-06	操作系统	0	不及格
共 11 条 10条/页 < 1 2 > 前往 1 页			

2. 管理员界面

- 查询考试信息

考试管理									
试卷名称	介绍	所属学院	所属专业	年级	考试日期	持续时间	总分	试卷类型	操作
计算机网络	2022年上期期末考试	软件工程学院	计算机科学与技术	2022	2022-04-21	120	100	期末考试	编辑 删除
数据库理论	2022年上期期末考试	软件工程学院	网络工程	2022	2022-04-17	90	100	期末考试	编辑 删除
数据结构	2022年下期期末考试	软件工程学院	软件工程	2022	2022-04-17	90	100	期末考试	编辑 删除
计算机导论	2022年上期期末考试	软件工程学院	计算机科学与技术	2022	2022-04-14	90	100	期末考试	编辑 删除
共 8 条 4条/页 < 1 2 > 前往 1 页									

- 添加试题

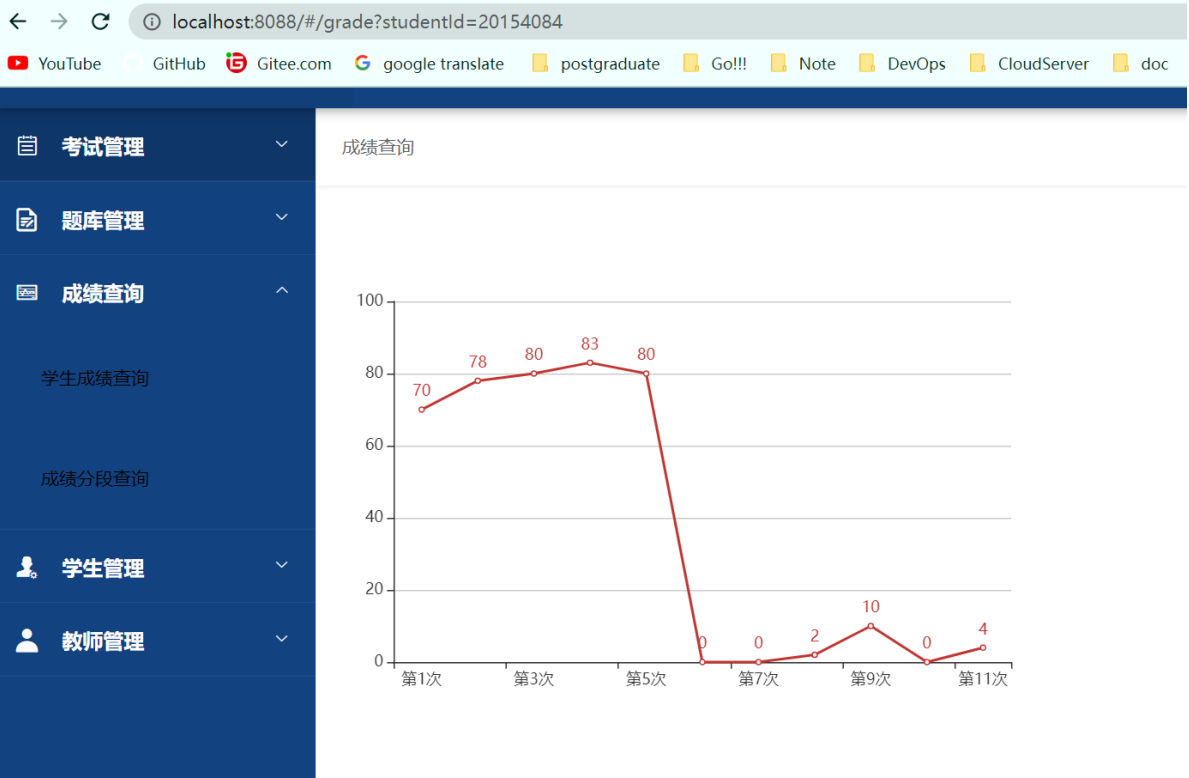
<div>考试管理</div> <div>题库管理</div> <div>功能介绍</div> <div>所有题库</div> <div>增加题库</div> <div>成绩查询</div> <div>学生管理</div> <div>教师管理</div>	<div>题库管理</div> <div><div>添加试题</div><div>在线组卷</div></div> <div><div>题目类型:</div><div>所属章节:</div><div>难度等级:</div><div>正确选项:</div></div> <div><div>题目:</div><div>请在下面的输入框中输入题目,形如--DNS 服务器和DHCP服务器的作用是 ()</div><div><div>请输入题目内容</div></div><div><div>A</div><div>请输入选项A的内容</div></div><div><div>B</div><div>请输入选项B的内容</div></div><div><div>C</div><div>请输入选项C的内容</div></div><div><div>D</div><div>请输入选项D的内容</div></div><div><div>解析:</div><div>请在下面的输入框中输入题目解析</div><div><div>请输入答案解析</div></div></div><div>立即添加</div></div>
---	--

查看学生成绩

成绩查询

姓名	学院	专业	年级	班级	性别	联系方式	查看成绩
王二虎	软件工程学院	计算机科学与技术	2019	2	男	13585439532	查看成绩
李林飞	软件工程学院	计算机科学与技术	2019	2	男	13658377857	查看成绩
王文林	软件工程学院	信息工程	2020	1	男	15583829425	查看成绩
linfei li	软件	软件	2019	1	男	18387400236	查看成绩

共 4 条 6条/页 < 1 > 前往 1 页



学生管理

姓名	学院	专业	年级	班级	性别	联系方式	操作
王二虎	软件工程学院	计算机科学与技术	2019	2	男	13585439532	编辑 删除
李林飞	软件工程学院	计算机科学与技术	2019	2	男	13658377857	编辑 删除
王文林	软件工程学院	信息工程	2020	1	男	15583829425	编辑 删除
linfei li	软件	软件	2019	1	男	18387400236	编辑 删除

共 4 条 6条/页 < 1 > 前往 1 页

教师管理

考试管理

题库管理

成绩查询

学生管理

教师管理

教师管理

添加教师

教师管理

姓名

学院

性别

电话号码

密码

身份证号

职称

立即创建

取消

3. 教师界面

XIAOJI-EXAM

张林

考试管理

题库管理

成绩查询

学生管理

很高兴遇见你,张林老师。

教务公告:

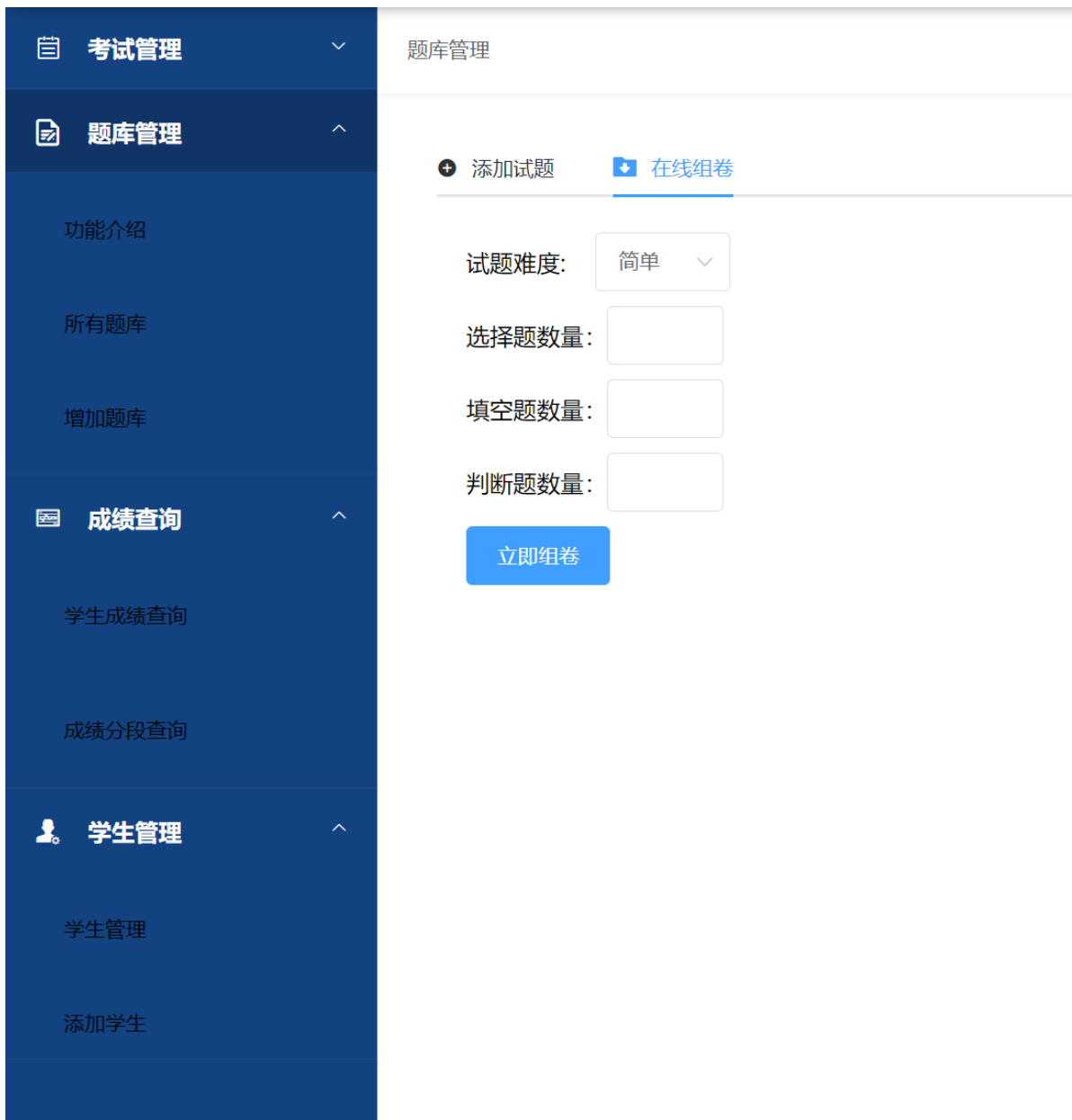
疫情防控期间本科生教学相关业务FAQ

疫情防控期间本科生教学相关业务FAQ

疫情防控期间本科生教学相关业务FAQ

疫情防控期间本科生教学相关业务FAQ

- 在线组卷：根据题型比例，查询数据库中的题库进行随机组成



3. 系统运行

3.1 前端运行

-开发环境：VS Code + Node.js

- 在Vue-cli环境下,安装依赖

```
1 | npm install
```

- 运行

```
1 | npm run dev
```

- 打包

```
1 npm run build
```

- 你可以点击【测试账号查看详细信息，默认是学生登录】



版权所有 ©2022 同济大学软件学院李林飞 保留所有权利 ICP备19001371号

3.2 后端运行

- 开发环境：IDEA 2021 + Maven 3.8
- 执行 `./sql` 目录下的脚本，创建数据库 `xiaoji_exam`
- 在 `application.properties` 配置文件中更改连接数据库的地址

```
1 spring.datasource.username=root
2 spring.datasource.password=123456
3 spring.datasource.url=jdbc:mysql://localhost:3306/xiaoji_exam?
  useUnicode=true&characterEncoding=UTF-8&useSSL=false&serverTimezone=UTC
4 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
5 spring.datasource.type=com.alibaba.druid.pool.DruidDataSource
```

- 启动运行

4. 实验总结

通过本次实验，掌握了以下技术：

- SpringBoot + MyBatisPLUS + MySQL + Swagger进行后端开发
- Vue + VueRouter + VueCookies + Vuex + ElementUI进行前端开发
- axios进行前后端的交互
- 通过路由控制实现不同角色展示不同的界面

改进方面：

- 可以使用token进行认证登录。
- 在线考试系统比较单调，部分功能还没有实现。

