报告内容

1. Github项目地址

<https://github.com/whu-iss/EasyExam>

1. PSP （估计将在程序的各个模块的开发上耗费的时间）

|  |  |  |
| --- | --- | --- |
| PSP2.1 | Personal Software Process Stage | Time/min |
| Planning | 计划 | 30 |
| ·Estimate | ·估计这个任务需要多长时间 | 10 |
| Development | 开发 | 120 |
| ·Analysis | ·需求分析 | 10 |
| ·Design Spec | ·生成设计文档 | 5 |
| ·Design Review | ·设计复审 | 5 |
| ·Coding Standard | ·代码规范 | 6 |
| ·Design | ·具体设计 | 15 |
| ·Coding | ·具体编码 | 160 |
| ·Code Review | ·代码复审 | 20 |
| ·Test | ·测试 | 30 |
| Reporting | ·报告 | 10 |
| ·Test Report | ·测试报告 | 5 |
| ·Size Measurement | ·计算工作量 | 10 |
| ·Postmortem&Process | ·事后总结，提出过程 | 10 |

1. 解题思路

刚开始拿到题目时，两个人首先分享对于做奥数（小学）试卷生成系统的看法，确定最终需求并设计各个页面。通过对现实考试模式和备考方案的调研，对模拟现实生活中的做奥数（小学）试卷考试系统的功能进行大概描述，主要分为前后台的管理。前台提供一个接口供学生登录网站参加考试或练习，后台提供一个接口供管理员或教师对网站进行考务维护和题库以及试卷的管理：其中前台需要实现在线考试、随机组卷、客观题自动评分、快速考试、继续考试、参加培训、错题训练等功能；后台的管理员或者教师拥有考务审核、人工阅卷、考试成绩统计、学习记录查询、题库管理、培训、虚拟班级、个人信息维护等功能。

通过查阅书籍，上网寻找资料，分工合作完成最终系统。当出现分歧时学会换位思考，通过深入探讨找到分歧中的共同点，取长补短，最终达成一致意见。

1. 设计实现过程

为了更好地对系统的构建进行管理，本系统引入了基于项目对象模型的构建工具Maven，使用父子项目的方式对系统进行了构建，系统框架分为四个模块，分别是common、portal、management、scoreMarker。其中，common作为系统的通用类库，包含系统前、后台的PO持久对象和POJO简单Java对象，通用工具类以及Constant常量类，这个module作为portal和management的通用类库。portal、management分别作为系统的前台项目、后台项目，独立运行。scoreMarker作为系统的计分项目，使用apache\_commons\_daemon将java程序做成系统服务，接收RabbitMQ队列中的答题卡并计分。

1. 代码说明。展示展示每个功能的核心代码，并解释思路与注释说明。

### 5.1 登录注册模块

**5.1.1部分代码截图展示**，如图5-1、5-2所示：

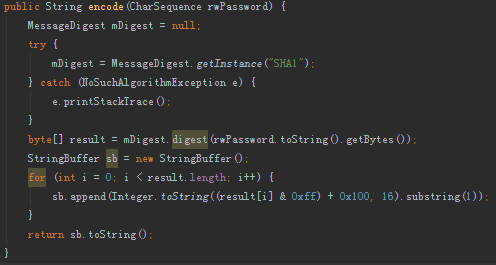


图5-1 StandardPasswordEncoderForSha1的encode方法

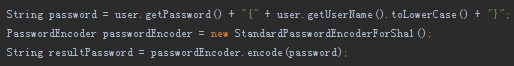


图5-2 调用encode方法进行加密

### 5.1.3 简要说明

以上为系统登录注册模块的部分代码以及页面展示，其中后台管理员和教师用户不提供注册页面，仅能通过系统设定的默认管理员账号从后台管理系统中添加。系统设计的用户角色有三类：超级管理员、教师、学生。角色注册功能中用户密码加密使用的是sha1+salt加密算法，salt使用的是用户名（若包含英文全部转小写），加密规则为取password+‘{’+username+‘}’ 字符串后，然后对其进行sha1加密。

### 5.2 实现考试功能

### 5.2.1 代码展示

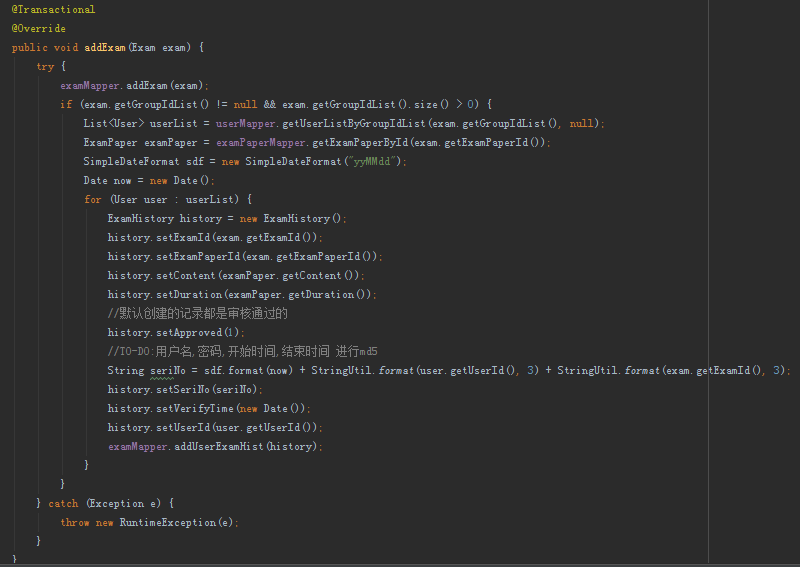


图5-6 ExamServiceImpl

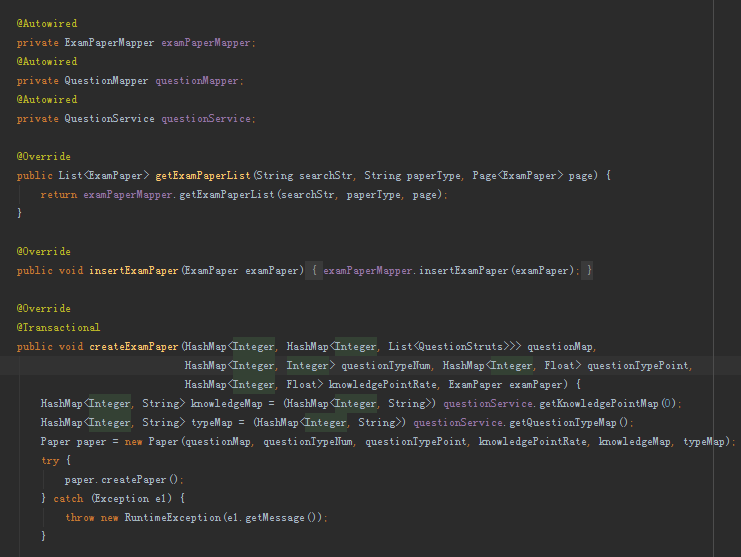


图5-7 ExamPaperServiceImpl

### 5.4.3 简要说明

考试功能依赖试题管理、试卷管理等功能。想进行一场考试，首先需要增加试题，然后对试题进行手动或自动组卷，最后选择相应试卷发布考试。考试分公、私有、模拟考三种。在发布的考试中可以查看相应学生名单，其中包含准考证号，学生可以通过此准考证号免登陆快速参加考试。此外，考试中断后在有效时间内重新进入考试可以恢复到上一次进行状态，这个状态以json形式保存在localstorage中，同时答题卡功能也是一下亮点设计，做过的题目在答题卡上会以灰色背景标注，学生答题时可以标注某题不确定，对应的题目在答题卡上会有蓝色三角上标提示。

## *5.3 实现审核与统计功能*

### 5.3.1 代码展示

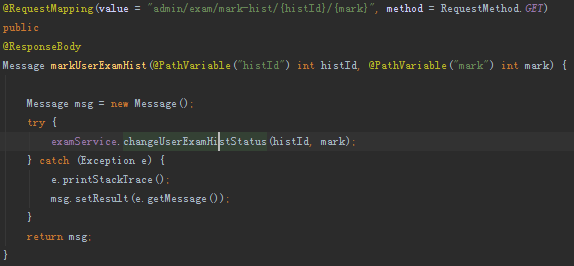


图5-9 改变用户考试审核状态

### 5.5.3 简要说明

审核功能主要是对用户申请考试、阅卷以及留言内容进行审核，核心是调用一个markXXX方法改变对应实体的一个标志位并对其所在表执行一个update操作。统计功能主要是对用户练习过的未做题目、错题、考试历史以及培训历史进行统计，数据信息主要通过联合查询获取，然后将这些数据作为源，利用mirror和highchart进行绘图，实现数据可视化。

## *5.4 实现练习功能*

### 5.4.1 代码展示

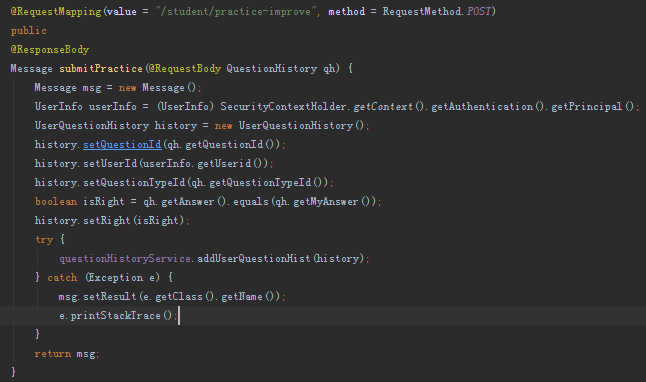


图5-12 练习功能

### 5.4.2 简要说明

练习功能包含强化练习、错题练习和随机练习。强化练习是对某个分类题库全部题目进行练习，错题是查询et\_user\_question\_history表中当前用户is\_right为0的question\_id，随机抽题即调用Random类的nextInt方法从题库中抽取指定记录条数的题目。进入练习后，又分为答题模式和背题模式，答题模式点击提交答案后如果正确会跳到下一题继续答题，如果错误会提示答案；背题模式开启后将无法答题，会将答案和解析显示出来仅供学生浏览，通过css的display属性进行实现。

## *5.5 实现培训功能*

### 5.5.1 代码展示

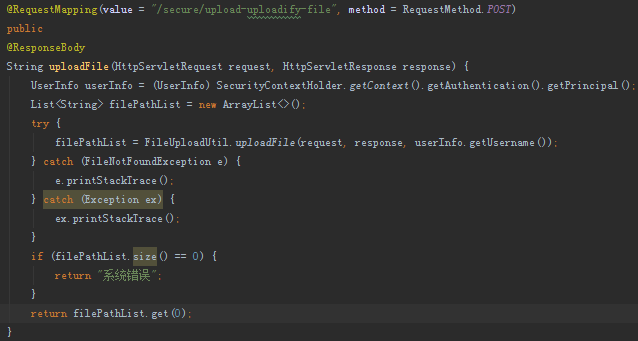


图5-13 培训内容上传

### 5.5.2 简要说明

前台学生参加培训观看视频和阅读pdf等内容主要由video.js和html5的video、embed标签以及对http头信息中的MIME类型的控制实现，后台培训上传文件用到了spring的FileCopyUtils以及jdk的Servlet、I/0等组件实现。

1. 测试运行。

### 界面展示

1. 前台学生注册，如图5-3所示：



图5-3 前台学生注册图

1. 前台学生登录，如图5-4所示：

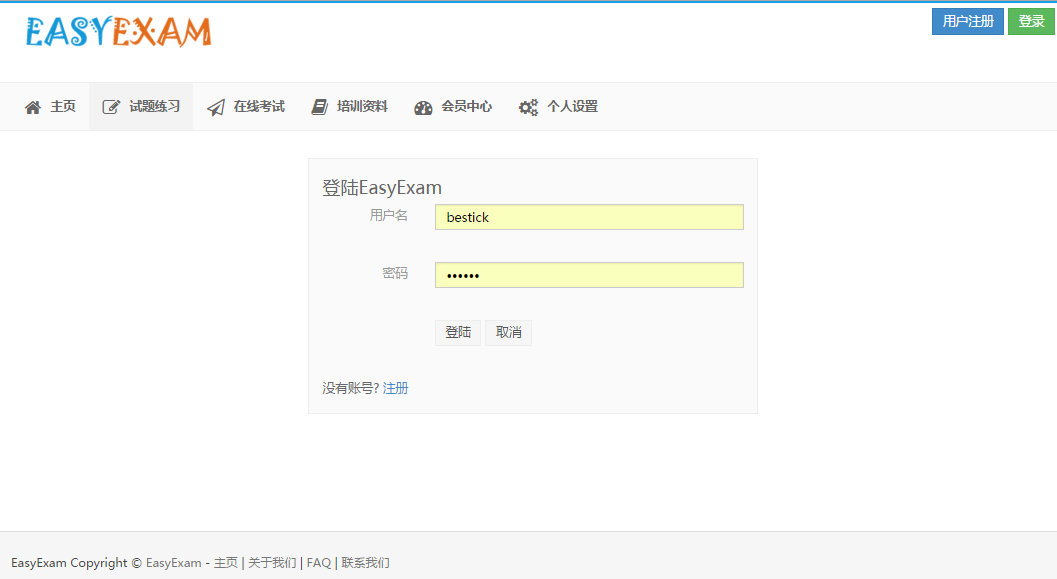


图5-4 前台学生登录图

1. 易考后台管理系统登录界面，如图5-5所示：



图5-5 易考后台管理系统登录界面

（4）实现审核与统计功能



图5-10 学生申请考试



图5-11 后台进行审核

## （5）实现考试功能



图5-8 发布考试截图

1. 合作情况

开始合作时，首先根据题目要求的考试系统需求的初步描述，对现实考试模式的需求进行分析，将系统的用户角色主要可以划分为超级管理员，教师用户和学生用户三类。

在扮演不同角色、角色切换时对于需求分析角度会有明显区别，我们发现需求本身具有不确定性，同时会随着开发的过程持续变化。通过合作讨论的方式，不仅仅可以将关键性的需求发掘出来，同时对于阐述不一致的地方，不一定代表对需求理解的对和错，相反，这种阐述的多样性有助于我们应对将来可能会发生的需求变化。我们结合现实中的调查，将需求进行精炼、分析与检查得到最终的意见。

1. 项目小结。

从拿到题目到项目结束，按照软件工程软件开发的流程对系统进行设计开发，实现中遇到了许许多多的问题，与最开始预估的时间安排有所出入，但是大致上达到要求。在遇到问题，分析问题和解决问题的过程中，网站也慢慢地成型，虽然在功能的细节上还是比较粗糙，但总体上还是达到了当初的设计要求。对于系统的通用性还需要再应对不同行业的在线考试模式进行统计分析。对于目前的出卷功能，有随机组卷和人工组卷方式，考虑到现实考试中对知识点分布有较高要求，后期系统需要对随机组卷中加入知识点定制后再随机组卷功能。通过本次项目使自己的开发技术和动手实践能力上成熟了很多，是今后开发项目的宝贵经验。