

Java 程序设计在线考试系统的开发

蒋凌志

(苏州高博软件技术职业学院, 江苏 苏州 215000)

摘要: 随着计算机应用技术与互联网技术的迅猛发展, 教学活动的各个环节也在向互联网上迁移。分析了考试的流程。用SSM框架技术实现了Java程序设计在线考试系统, 服务器用的是Tomcat, 数据库用的是MySQL。该系统充分使用了工厂设计模式, 面向接口的编程, 自动装配加载, 注解, Mybatis等技术, 使得开发更方便, 后期维护更便捷, 系统运行效率比较高。

关键词: 在线考试; SSM; JavaWeb; Spring; SpringMVC; Mybatis

中图分类号: TP311.1 文章标识码: A

文章编号: 1009-3044(2019)07-0142-03

DOI: 10.14004/j.cnki.ckt.2019.1013

Development of Java Programming Online Examination System

JIANG Ling-zhi

(Global Institute of Software Technology, Suzhou 215000, China)

Abstract: With the rapid development of computer application technology and Internet technology, all aspects of teaching activities are also migrating to the Internet. The process of examination is analyzed. The online test system of Java programming is implemented with SSM framework technology. Tomcat is used in the server and MySQL is used in the database. The system makes full use of the technology of factory design mode, Interface-oriented programming, automatic assembly and loading, annotation, Mybatis and so on. It makes the development more convenient, the later maintenance more convenient, and the system runs more efficiently.

Key words: online examination; SSM; javaweb; spring; springMVC; mybatis



开放科学(资源服务)标识码(OSID):

1 背景

现阶段, 学校中各种考试大都采用传统的考试方式。在此方式下, 组织一次考试至少要经过五个步骤, 即人工出卷, 考生考试, 人工阅卷, 成绩评估和试卷分析。但随着计算机应用的迅猛发展, 网络应用不断扩大, 如远程教育和虚拟大学的出现等等, 人们也利用这些技术来进行在线考试, 以减轻教师的工作负担及提高工作效率, 与同时也提高了考试的质量, 从而使考试更趋于公证、客观、更加激发学生的学习兴趣。例如目前许多国际著名的计算机公司所举办的各种认证考试绝大部分采用这种方式。为了适应新形势的发展, 推出了Java程序设计在线考试系统, 使其发挥高效、便捷的作用, 把老师从繁重的工作中解脱出来。^[1-4]

2 技术选取

现在流行的SSM框架是Spring MVC, Spring和Mybatis框架的整合, 是一种标准的MVC模式, 这个框架将整个系统划分为View表现层, Controller层, Service层, DAO层四层。使用Spring MVC负责请求的转发和视图管理, Spring实现业务对象管理, Mybatis作为数据对象的持久化引擎。^[5-8]

SpringMVC^[6]:

- 1) 客户端发送请求到DispatcherServlet(分发器)。
- 2) 由DispatcherServlet控制器查询HandlerMapping, 找到处理请求的Controller。
- 3) Controller调用业务逻辑处理后, 返回ModelAndView。
- 4) DispatcherServlet查询视图解析器, 找到ModelAndView指定的视图。
- 5) 视图负责将结果显示到客户端。

Spring^[7]: 平时开发接触最多的容器就是IOC容器, 它可以自动加载bean类, DAO类, Service类, 有了这个机制, 就不用每次使用这个类的时候为它初始化。另外Spring的aop, 事务管理等等都是经常要用到的。

Mybatis^[8]: Mybatis是对JDBC的封装, 它让数据库底层操作变的透明。Mybatis的操作都是围绕一个sqlSessionFactory实例展开的。Mybatis通过配置文件关联到各实体类的Mapper文件, Mapper文件中配置了每个类对数据库所需进行的SQL语句映射。在每次与数据库交互时, 通过sqlSessionFactory拿到一个sqlSession, 再执行SQL命令。

所以, 整个框架的流程如图1所示:

收稿日期: 2019-01-19

作者简介: 蒋凌志(1977—), 男, 江苏苏州人, 讲师, 硕士, 主要研究方向为软件技术。

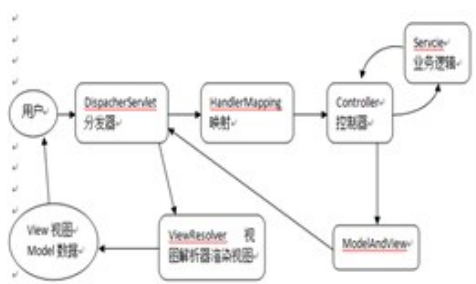


图1 SSM框架流程图

3 系统设计

3.1 总体设计

3.1.1 设计需求

1)在线考试系统可以帮助教师完成一个考试从题目设计,考试安排,考试实施,考卷批改到分数统计总结的所有工作。

2)所有的考试数据与试卷数据需要用数据库技术进行存储和管理^[3]。

3)考试数据的存放和处理必须对考试保密,需要一定的安全性保障^[2]。

4)这个系统是分布式的,这意味着只要有考试客户端可以连接到考试应用服务器,任何考生在任何地方进行身份确认,都能完成考试^[3]。

5)题目最好有一定的稳定性和随机性。稳定性可以保证每一次考试对每一个考生是公平的,随机性可以避免作弊的发生^[2]。

6)考试的部分结果在考生考完之后就可以知道,同时提供一个机会让考生知道自己哪里做错了,并给出反馈信息说明理由。

根据以上需求,做出如图2所示的流程图与图3所示的系统设计图。

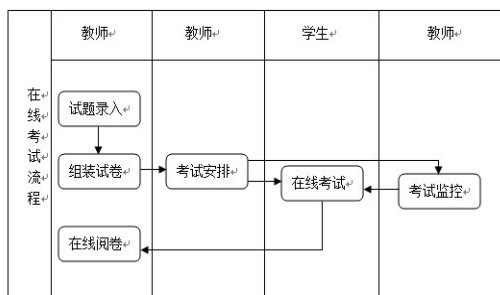


图2 在线考试流程图

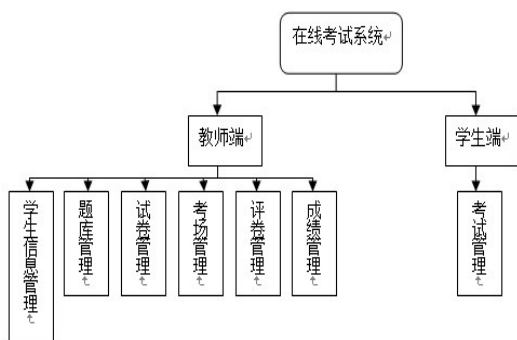


图3 系统设计图

学生信息管理:对学生的学号,姓名,班级,课程等数据进行增删改查操作。

题库管理:对课程的题目进行管理,对题目的难度,类型,分值,答案,所属章节等数据进行增删改查操作。

试卷管理:从题库中按不同模式进行抽题组卷操作,作废试卷,修改试卷操作。

考场管理:对选中的班级,抽中的试卷进行开考操作,记录时间,延长时间,指定交卷,到时自动交卷操作。

评卷管理:对客观题进行自动评卷,主观题显示答题要点与学生的作答,不显示学生信息,由教师评卷。

成绩管理:教师对考试成绩进行分析统计和察看。

3.2 数据库设计

根据上面的系统设计,列出所需的数据,设计出如图4所示的数据库设计图。

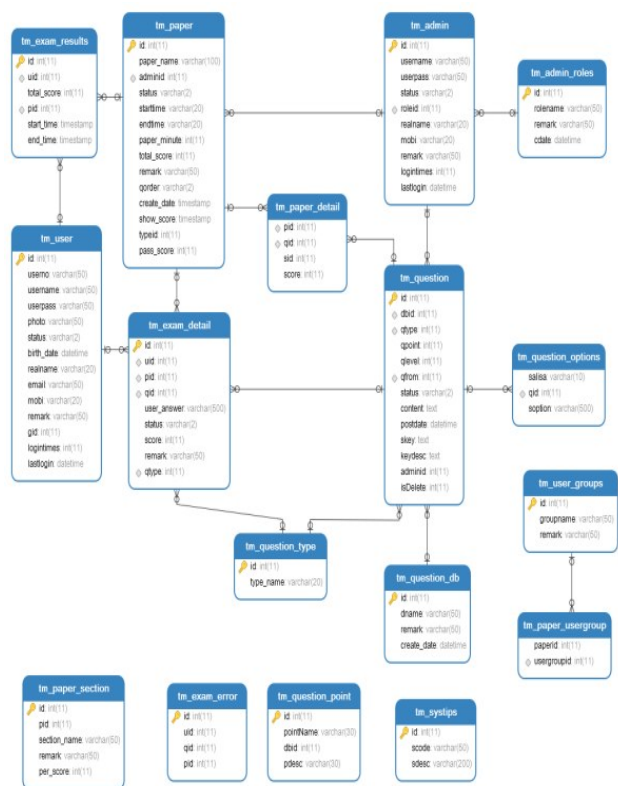


图4 数据库设计图

3.3 详细设计

本节以用户注册为例介绍。

1)先对总体配置文件web.xml进行配置。

配置前端控制器如图5所示:

```

<!-- 配置前端控制器 -->
<servlet-name>springMVC</servlet-name>
<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
<init-param>
<param-name>contextConfigLocation</param-name>
<param-value>classpath:applicationContext.xml</param-value>
</init-param>
  
```

图5 配置前端控制器

2)设置SpringMVC的配置文件applicationContext.xml:

配置如图6所示:

```

<!-- 配置视图解析器 -->
<bean
    class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/" />
    <property name="suffix" value=".jsp" />
</bean>
<!-- 设置数据库连接文件 -->
<bean
    class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">
    <property name="location" value="classpath:datasource.properties" />
</bean>
<!-- 定义数据源 -->
<bean name="dataSource" class="org.apache.commons.dbcp.BasicDataSource">
    <property name="driverClassName" value="${driver}" />
    <property name="url" value="${url}" />
    <property name="username" value="${username}" />
    <property name="password" value="${password}" />
</bean>
<!-- 定义SessionFactory -->
<bean name="sessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource" />
    <property name="configLocation" value="classpath:mybatis.xml" />
    <property name="mapperLocations">
        <list>
            <value>classpath*:com/kxx/entity/*.xml</value>
        </list>
    </property>
</bean>
<!-- 定义DAO层 -->
<bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">
    <property name="basePackage" value="com.online.dao" />
    <property name="sqlSessionFactoryBeanName" value="sessionFactory" />
</bean>

```

图6 配置SpringMVC

设置自动扫描有Spring注解的类Controller层,Service层:

```
<context:component-scan base-package="com.online.controller"/>
```

该配置会自动扫描所有在 com.online.controller 包中有Spring注解的Controller类,并自动进行装配注入。

```
<context:component-scan base-package="com.online.service"/>
```

该配置会自动扫描所有在 com.online.service 包中有Spring注解的Service类,并自动进行装配注入。

3)设置用户管理的POJO类:

```

public class TmUser implements java.io.Serializable {
    private Integer id;
}

```

4)设置用户管理的Mybatis的映射文件TmUser.xml如图7所示:

```

<mapper namespace="com.online.dao.TmUserMapper">
<!-- 用户注册 -->
<insert id="userRegister" parameterType="TmUser">
    INSERT INTO `tomexam`.`tm_user` (
        <trim suffixOverrides=",">
            <if test="id != null and id != ''">id,</if>
            省脑.....
        </trim>
    )
    VALUES
    (
        <trim suffixOverrides=",">
            <if test="id != null and id != ''">#{id},</if>
            省脑.....
        </trim>
    );
</insert>
</mapper>

```

图7 TmUser.xml配置文件

该Mapper中定义了一个注册用户的数据库操作,使用到了Mybatis的动态SQL技术,使用if来控制字段与值的添加和取消。可以根据条件来自动生成SQL语句。

5)设置请求视图register.jsp

图8 用户注册页面

```

@ResponseBody
@RequestMapping("register.do")
public String register(TmUser tmUser,String u_usercode,
String t,HttpSession session) {
    Map<String, String> date = new HashMap<String,
String>();
    if (tmUserService.userRegister(tmUser)>0) {
        date.put("code", "1");
        date.put("message", "注册成功");
    }else{
        date.put("code", "0");
        date.put("message", "注册失败");
    }
    return JSONArray.toJSONString(date);
}

```

在register方法上进行注解,响应register.do的请求,在方法中调用了前面已配置好的注解的注册服务,得到结果。整个调用过程非常流畅简洁,这就是Spring的魅力所在。

4 结束语

首先提出了设计需求,对考试流程进行分析,根据需求与流程进行了总体设计,进而数据库设计。用SSM框架实现了一个Java程序设计在线考试系统。当然把题库改了,可以应用于其他科目的考试。这个系统可以大大减轻教师的工作负担及提高工作效率,与此同时也提高了考试的质量,从而使考试更趋于公证、客观、更加激发学生的学习兴趣。由于使用了SpringMVC技术,使用接口编程,所以后期的维护和更新会比较方便,可以像更换机器零件一样更新服务,实现新功能。

参考文献:

- [1] 陈荟慧, 张新颜. 在线考试系统的分析与设计[J]. 计算机技术与发展, 2009, 19(10): 214-217.
- [2] 武伟, 魏晓. 在线考试作弊防御方法的设计与实现[J]. 上海应用技术学院学报: 自然科学版, 2006, 6(1): 49-51.
- [3] 唐俊武, 南理勇, 左强. 在线考试系统开发中的几个问题及解决方法[J]. 计算机与数字工程, 2005, 33(8): 144-147.
- [4] 叶溪溪, 吴观茂. 在线考试系统分析与设计[J]. 电脑知识与技术, 2016, 12(3): 104-106.
- [5] 宋雪亚, 王传安. 基于JSP的在线考试系统设计与实现[J]. 湖南工程学院学报: 自科版, 2017, 27(2): 33-38.
- [6] 李洋. SSM框架在Web应用开发中的设计与实现[J]. 计算机技术与发展, 2016, 26(12): 190-194.
- [7] 邹红霞. 基于SSM框架的Web系统研究与应用[J]. 湖南理工学院学报: 自科版, 2017, 30(1): 39-43.
- [8] 王鹤琴, 汪炜玮, 朱珍元. 基于SSM框架技术的办公管理系统的研究[J]. 安徽警官职业学院学报, 2017, 16(3): 124-128.

【通联编辑: 谢暖暖】